CICS® Transaction Server for OS/390®

CICS Supplementary Data Areas

Release 3

LY33-6090-02

CICS® Transaction Server for OS/390®

CICS Supplementary Data Areas

Release 3

“Restricted Materials of IBM”
Licensed Materials – Property of IBM

LY33-6090-02

Note!

Before using this information and the product it supports, be sure to read the general information under “Notices” on page vii.

Third edition (March 1999)

This edition applies to Release 3 of CICS Transaction Server for OS/390, program number 5655-147, and to all subsequent versions, releases, and modifications until otherwise indicated in new editions. Make sure you are using the correct edition for the level of product.

The CICS Transaction Server for OS/390 Release 2 edition remains applicable and current for users of CICS Transaction Server for OS/390 Release 2, and may be ordered using its order number, LY33-6090-01.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

At the back of this publication is a page entitled “Sending your comments to IBM”. If you want to make comments, but the methods described are not available to you, please address them to:

IBM United Kingdom Laboratories, Information Development,
Mail Point 095, Hursley Park, Winchester, Hampshire, England, SO21 2JN.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 1977, 1999. All rights reserved.**
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Notices vii
Trademarks viii

Preface ix
CICS Transaction Server for OS/390 ix
  CICS books for CICS Transaction Server for OS/390 ix
  CICSPlex SM books for CICS Transaction Server for OS/390 x
  Other CICS books x

Chapter 1. CTS 390 1.3 Supplementary Data Areas 1
How the data areas are presented 1
APIQ Inquire application data xpi command 2
APLI Language interface work area 3
BAACT Bam process class 5
BAACT Bam contaier_set class 9
BAACT Bam activity class 10
BAACT Bam container class 21
BAAR Bam audit record class 22
BAPT Bam processtype class 23
BRDCC Bridge control blocks 24
CAUTR CICS affinities utility trace table 26
CCGD Catalog static storage 29
CPCPS Cpi-c conversation control block 32
CPSPS Cpi static storage area 34
DDBSC Directory manager building blocks 35
DDCBC Directory manager structures 36
DEGPC Dce services domain global statistics 38
DHANC Document handler anchor block 39
DHTL Document handler template descriptor 43
DMAFC Dm authorised facility state 45
DMCB1 Domain manager anchor block 47
DMCB2 Domain manager browse cursor 49
DMCB3 Domain manager wait queue element 50
DMCB4 Domain record 51
DMENC Domain manager enf state 52
DSANC Dispatcher domain anchor block 53
DSTBA Task browse area 63
DSTSK Dispatcher domain task description 64
DTCPS Data tables local access anchor blocks 68
DTLPS Data tables connection anchor blocks 69
DTRPS Data tables remote sharing anchor block 72
DTSPS Data tables SVC routine anchor blocks 72
DTXPS Data tables security anchor block 74
DUFC Dump formatting communication area 75
DUFP Parameter area declarations 76
D2CSB Csub block 78
D2ENT Db2entry block 81
D2GLB Cics/db2 global block 85
D2GWA Cics/db2 global work area 92

D2LOT Cics/db2 life of task block 93
D2SS Cics/db2 static storage 96
D2TRN Db2tran block 98
FBWAC File browse work area for data tables 99
FCPEC File control cfdt pool element 101
FCPWC File control cfdt pool wait element 102
FCQRE File control quiesce receive element 104
FCQSE File control quiesce send element 105
FCUPC File control cfdt uow pool block 107
FEP01 Frontend programming interface trace 108
FEP02 Adapter resource manager 113
FEP03 VTAM acb work area 115
FEP04 Bind request save area 116
FEP05 Connection descriptor 117
FEP06 Common data area 120
FEP07 Conversation data area 125
FEP08 Device support extension 127
FEP09 Tsf - eye catcher map 131
FEP10 Node descriptor 132
FEP11 Pool descriptor 134
FEP12 Properties list 135
FEP13 Property set info 136
FEP14 Work queue element 138
FEP15 VTAM receive request block 139
FEP16 VTAM requests block 140
FEP17 Request parameter area 141
FEP18 Session control request block 145
FEP19 Terminal simulation facility 146
FEP20 Target descriptor 147
FEP21 Frontend programming interface 148
FLLBC File control locks locator block 150
KCB Kernel anchor block 151
KECB Kernel control blocks 155
KEMHD Kernel module header 161
KESTP Kernel stack entry 163
LDCBS Loader domain control blocks 164
LGANC Logger domain anchor block 188
LGFL Log of logs failure record 198
LGSF System log format 199
LIFO Stack segment table header 203
LMCB1 Lock manager domain anchor block 204
LMCB2 Lock manager domain quickcell headers 206
L2BL Log manager block class 208
L2BS Log manager browseable stream class 211
L2CH Log manager chain class 219
L2DM Log manager l2dm class 224
L2HP Log manager history point class 226
L2HS Log manager hard stream class 227
L2LF Log manager log formats 231
L2LT Log manager lock tracker class 238
L2RT Log manager record token class 239
L2SL Log manager system log class 240
L2SR Log manager stream class 242

MEMMS Message table definition 252
MEPS Message domain anchor block 257
MNAFB Monitoring authorised parameter block 260
MNCBS Monitoring domain control blocks 262
NQA Enqueue domain anchor block 275
NQB Enqueue domain browse element 276
NQEA Enqueue domain queue element area 277
NQOX Enqueue domain browse owner extension 279
NQPL Enqueue domain enqueue pool 280
NQWX Enqueue domain browse waiter extension 282
PAA Parameter manager domain anchor block 283
PGA Macro save area 285
PGDCC Program manager control blocks 286
PGHM Handle manager declarations 293
PRS Partner domain static storage area 296
PTE Partner table entry 297
RDAB Resource definition anchor block 299
RDUB Resource definition update block 300
RMDM Recovery manager domain management instance 301
RMID Recovery manager identity instance 303
RMLI Recovery manager loggable object identity 304
RMLK Recovery manager link class data 305
RMLK Recovery manager link instance 309
RMLS Recovery manager link set instance 318
RMNM Recovery manager logname class data 320
RMNM Recovery manager logname instance 321
RMNS Recovery manager logname set instance 322
RMRO Recovery manager resource owner instance 324
RMSL Recovery manager system log instance 327
RMSL Recovery manager system log class data 329
RMUW Recovery manager unit of work instance 330
RMUW Recovery manager unit of work class data 337
RRAB Resource definition recovery definitions 341
RUEI Logger reusable extended iliffe vector class 343
SHRTC Sh request routing class 344
SMDCC Storage manager anchor block 345
SMMCC Sm macro-compatability anchor block 364
SOA Sockets anchor block 366
STAFB Statistics authorised parameter block 373
STCB1 Statistics domain anchor block 374
STUCB Statistics utility program anchor block 375
TIA Timer domain anchor block 378
TSA Temporary storage anchor block 380
TSAUX Temporary storage auxiliary class 384
TSMN Temporary storage model class 390
TSMN Temporary storage main class 392
TSNM Temporary storage name class 393
TSOL Temporary storage ownership lock class 394
TSQU Temporary storage queue class 396
TSRL Temporary storage shared class 399
TSRL Temporary storage resource lock class 401
TSWQ Temporary storage wait queue class 402
UDB User domain user data block 403

USANC User domain anchor block 405
USGPS User domain statistics 409
USXD User domain transaction data 410
USXT User domain transaction token 411
WBABC Web anchor block 411
WBANC Web domain anchor block 412
WBA1C Web business logic compatibility interface 413
WBBLC Web business logic interface parameters 416
WBEPC Web error program parms 419
WBSTC Web state manager data 422
WBUCC Web interface urp constants 424
WRB Web request block class 427
XCCBC External CICS interface control blocks 431
XMANC Transaction manager domain anchor block 435
XMCAT Transaction manager catalog records 438
XMCLC Transaction manager transaction class 439
XMRLC Transaction manager resource lock element 440
XMXBC Transaction manager tran. browse element 441
XMXDC Transaction manager transaction definition 441
XMXNC Transaction manager transaction 445
XSANC Security domain anchor block 448
XSSS Security supervisor storage 451
XSXD Security domain transaction data 455
XSXT Security domain transaction token 456
ZCQ Builder services action blocks 456

Index 459

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply in the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore this statement may not apply to you.

This publication could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM United Kingdom Laboratories, MP151, Hursley Park, Winchester, Hampshire, England, SO21 2JN. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Programming License Agreement, or any equivalent agreement between us.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, or other countries, or both:

BookManager
C/370
CICS
CICS/ESA
CICS/MVS
DB2
IBM
IBMLink
LANGUAGE ENVIRONMENT
MVS/ESA
MVS/XA
OPENEDITION
OS/390
RACF
S/370
S/390
VTAM

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Preface

This manual is supplementary to the CICS® *Data Areas* manual. It contains data areas (control blocks, parameter lists and constants) that are part of the CICS product implementation. These data areas may be useful for tasks such as CICS problem diagnosis, performance monitoring, and tuning. These data areas are intended for use by only a limited set of users involved in designing products complementary to CICS that perform one of these specialized tasks and require this information, which can be expected to change with subsequent releases of CICS.

Most products can be designed without using the information provided by this manual, because they can use the facilities provided by the extended CICS API (for example, the EXEC CICS INQUIRE/SET commands), and the exit programming interface (XPI) provided by CICS.

This manual is not needed by CICS application programmers, nor is it required when requesting assistance from the IBM® Service organization.

Licensees are allowed to copy information derived from this manual into the source code of their products.

CICS Transaction Server for OS/390

CICS Transaction Server for OS/390: Planning for Installation	GC33-1789
CICS Transaction Server for OS/390 Release Guide	GC34-5352
CICS Transaction Server for OS/390 Migration Guide	GC34-5353
CICS Transaction Server for OS/390 Installation Guide	GC33-1681
CICS Transaction Server for OS/390 Program Directory	GC33-1706
CICS Transaction Server for OS/390 Licensed Program Specification	GC33-1707

CICS books for CICS Transaction Server for OS/390

General	
CICS Master Index	SC33-1704
CICS User's Handbook	SX33-6104
CICS Glossary (softcopy only)	GC33-1705
Administration	
CICS System Definition Guide	SC33-1682
CICS Customization Guide	SC33-1683
CICS Resource Definition Guide	SC33-1684
CICS Operations and Utilities Guide	SC33-1685
CICS Supplied Transactions	SC33-1686
Programming	
CICS Application Programming Guide	SC33-1687
CICS Application Programming Reference	SC33-1688
CICS System Programming Reference	SC33-1689
CICS Front End Programming Interface User's Guide	SC33-1692
CICS C^{++} OO Class Libraries	SC34-5455
CICS Distributed Transaction Programming Guide	SC33-1691
CICS Business Transaction Services	SC34-5268
Diagnosis	
CICS Problem Determination Guide	GC33-1693
CICS Messages and Codes	GC33-1694

CICS Diagnosis Reference	LY33-6088
CICS Data Areas	LY33-6089
CICS Trace Entries	SC34-5446
CICS Supplementary Data Areas	LY33-6090
Communication	
CICS Intercommunication Guide	SC33-1695
CICS Family: Interproduct Communication	SC33-0824
CICS Family: Communicating from CICS on System/390	SC33-1697
CICS External Interfaces Guide	SC33-1944
CICS Internet Guide	SC34-5445
Special topics	
CICS Recovery and Restart Guide	SC33-1698
CICS Performance Guide	SC33-1699
CICS IMS Database Control Guide	SC33-1700
CICS RACF Security Guide	SC33-1701
CICS Shared Data Tables Guide	SC33-1702
CICS Transaction Affinities Utility Guide	SC33-1777
CICS DB2 Guide	SC33-1939

CICSPlex SM books for CICS Transaction Server for OS/390

General	
CICSPlex SM Master Index	SC33-1812
CICSPlex SM Concepts and Planning	GC33-0786
CICSPlex SM User Interface Guide	SC33-0788
CICSPlex SM View Commands Reference Summary	SX33-6099
Administration and Management	
CICSPlex SM Administration	SC34-5401
CICSPlex SM Operations Views Reference	SC33-0789
CICSPlex SM Monitor Views Reference	SC34-5402
CICSPlex SM Managing Workloads	SC33-1807
CICSPlex SM Managing Resource Usage	SC33-1808
CICSPlex SM Managing Business Applications	SC33-1809
Programming	
CICSPlex SM Application Programming Guide	SC34-5457
CICSPlex SM Application Programming Reference	SC34-5458
Diagnosis	
CICSPlex SM Resource Tables Reference	SC33-1220
CICSPlex SM Messages and Codes	GC33-0790
CICSPlex SM Problem Determination	GC33-0791

Other CICS books

CICS Application Programming Primer (VS COBOL II)	SC33-0674
CICS Application Migration Aid Guide	SC33-0768
CICS Family: API Structure	SC33-1007
CICS Family: Client/Server Programming	SC33-1435
CICS Family: General Information	GC33-0155
CICS 4.1 Sample Applications Guide	SC33-1173
CICS/ESA 3.3 XRF Guide	SC33-0661

Chapter 1. CTS 390 1.3 Supplementary Data Areas

How the data areas are presented

The data areas are listed in alphabetical order of their shortened names. The shortened name usually, but not always, matches the first few characters of the data area name, disregarding the DFH prefix; for example DFHTCA is shortened to TCA. Some data areas are grouped together according to usage. If you do not find a data area under the expected short name, you should look in the table of contents or the index for the full name of the area or for the name of the macro or copy book that generates the area.

For each field in each data area, the following information is listed:

- The hexadecimal offset, in parentheses
- The data type and for bitstring values, the bit representation
- The length in bytes (decimal)
- The name (symbolic label)
- A brief description of the function

Where the name of a field is shown as an asterisk (`*`), the field is reserved.

Where bit settings are indicated, the symbolic labels that have been equated to the bit settings are given. These labels are used to refer to the numeric values in programs that use the data area, and are included in this book to help you understand the program listings. The offset given for one of these fields applies only to the symbolic label assigned to the field as a unit; it does not apply to the labels equated to bit settings (hex values).

Where a storage definition has a duplication factor, for example `DCREGS (16)`, the length of the field is the length of each element of the storage. The total length of the storage is this length multiplied by the duplication factor which is shown in parentheses after the name.

For EQUATE statements, the operand is shown in quotation marks in the description.

Use of the index

- All fields are listed in the index at the back of this book.
- Each field name listed in the index is followed by:
  - the hexadecimal offset of the field, shown in parentheses,
    - If the field name applies to a bit value, this is indicated by the word **BIT** in place of the hexadecimal offset.
  - the field length, shown in square brackets,
  - the short name of the area in which it appears,
  - and the page number.

Use the index to find where this book shows the field that you are seeking, in a Data Area. Don't use the index for anything else — for example, you will probably not find enough information in the index to diagnose a problem.

APIQ Inquire application data xpi command

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	64	DFHAPIQ_ARG	
(0)	CHARACTER	16	APIQ_HEAD	
(0)	HALFWORD	2	APIQ_PLISTLEN	
(2)	HALFWORD	2	*	
(4)	FULLWORD	4	APIQ_FORMAT_NO	
(8)	FULLWORD	4	APIQ_VERSION_NO	
(C)	BITSTRING	4	*	
	1...		APIQ_KERNHANDLE	
(C)	BITSTRING	3	*	
64 EXISTENCE BITS ONE PER KEYWORD IN KEYWORD ORDER				
(10)	BITSTRING	8	APIQ_EXISTENCE	
	1...		APIQ_FUNCTION_X	
	.1..		*	
	..1.		APIQ_RESPONSE_X	
	...1		APIQ_REASON_X	
	.... 1...		APIQ_EIB_X	
	.... .1..		APIQ_SYSEIB_X	
	.... ..1.		APIQ_TCTUA_X	
	1		APIQ_TCTUASIZE_ X	
(11)	1...		APIQ_TWA_X	
	.1..		APIQ_TWASIZE_X	
	..1.		APIQ_RSA_X	
	...1		APIQ_DSA_X	
ACTUAL KEYWORDS NOW FOLLOW WITH THEIR RESPECTIVE ENUMERATED TYPES COMMENTED				
(18)	UNSIGNED	1	APIQ_FUNCTION	
	APIQ_ INQ_APPLICATION_ DATA CONSTANT(001)			
(19)	CHARACTER	1	*	
(1A)	UNSIGNED	1	APIQ_RESPONSE	
	APIQ_ OK CONSTANT(001) APIQ_ EXCEPTION CONSTANT(002) APIQ_ DISASTER CONSTANT(003) APIQ_ INVALID CONSTANT(004) APIQ_ KERNERROR CONSTANT(005) APIQ_ PURGED CONSTANT(006)			
(1B)	UNSIGNED	1	APIQ_REASON	
	APIQ_ DPL_PROGRAM CONSTANT(001) APIQ_ NO_TRANSACTION_ ENVIRONMENT CONSTANT(002) APIQ_ TRANSACTION_ DOMAIN_ERROR CONSTANT(003) APIQ_ INVALID_ FUNCTION CONSTANT(004) APIQ_ ABEND CONSTANT(005) APIQ_ LOOP CONSTANT(006) APIQ_ INQ_FAILED CONSTANT(007)			
(1C)	ADDRESS	4	APIQ_EIB	
(20)	ADDRESS	4	APIQ_SYSEIB	
(24)	ADDRESS	4	APIQ_TCTUA	
(28)	UNSIGNED	4	APIQ_TCTUASIZE	
(2C)	ADDRESS	4	APIQ_TWA	
(30)	UNSIGNED	4	APIQ_TWASIZE	
(34)	ADDRESS	4	APIQ_RSA	
(38)	ADDRESS	4	APIQ_DSA	
(3C)	CHARACTER	4	*	
(40)	CHARACTER		*	

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	APIQ_INQ_ APPLICATION_DATA	
1	DECIMAL	1	APIQ_OK	
1	DECIMAL	2	APIQ_EXCEPTION	
1	DECIMAL	3	APIQ_DISASTER	
1	DECIMAL	4	APIQ_INVALID	
1	DECIMAL	5	APIQ_KERNERROR	
1	DECIMAL	6	APIQ_PURGED	
1	DECIMAL	1	APIQ_DPL_PROGRAM	
1	DECIMAL	2	APIQ_NO_TRANSACTION_ ENVIRONMENT	
1	DECIMAL	3	APIQ_TRANSACTION_ DOMAIN_ERROR	
1	DECIMAL	4	APIQ_INVALID_ FUNCTION	
1	DECIMAL	5	APIQ_ABEND	
1	DECIMAL	6	APIQ_LOOP	
1	DECIMAL	7	APIQ_INQ_FAILED	

APLI Language interface work area

-

The Language Interface Work-Area is acquired by the Transaction Manager (XM) Domain during initial processing for the task. The area is built in the storage key defined by the TaskDataKey value of the Task definition.

If the length of this area changes, take great care to ensure that all modules affected either directly, or indirectly via DFHAPCOM or the change in length to language_interface_workarea, are re-compiled.

CONTROL BLOCK Name = DFHLIWAC
DESCRIPTIVE NAME = CICS Language interface Work Area
This Copy Book describes the common work area used for communications between CICS and the various run-time language environments.
FUNCTION = Interface between CICS and the Language Environments.
LIFETIME = Task
Storage CLASS = TaskDataKey.
LOCATION =
Addressed from the SYSTEM TCA by TCACEEPT.
Notes :
Dependencies = S/370
Restrictions =
Module Type = Control block definition

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	692	LANGUAGE_ INTERFACE_ WORKAREA	

The following area will hold the Thread Tokens used by Cobol II, C/370 and LE/370: and the addresses of any Thread work-areas acquired by use for C/370 or LE/370.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER	8	COBOL2_ THREAD_TOKEN	
(8)	CHARACTER	8	LE370_THREAD_ TOKEN	
(10)	CHARACTER	8	C370_THREAD_ TOKEN	
(18)	ADDRESS	4	LE370_THREAD_ WORKAREA_ADDR	
(1C)	ADDRESS	4	C370_THREAD_ WORKAREA_ADDR	

The next area is for the use of the Cobol II, C/370 and LE/370 Language Environment routines.

Offset Hex	Type	Len	Name (Dim)	Description
(20)	FULLWORD	4	LANG_ENV_ REASON_CODE	
(24)	CHARACTER	240	LANG_ENV_ WORKAREA	
(114)	FULLWORD	4	LANG_ENV_RSA (18)	

A save area to hold the values of the floating point registers at the time of an abend.

Offset Hex	Type	Len	Name (Dim)	Description
(15C)	CHARACTER	32	FLOATING_ POINT_REGISTERS	

Offset Hex	Type	Len	Name (Dim)	Description
(15C)	CHARACTER	8	FLOATING_ POINT_REG0	
(164)	CHARACTER	8	FLOATING_ POINT_REG2	
(16C)	CHARACTER	8	FLOATING_ POINT_REG4	
(174)	CHARACTER	8	FLOATING_ POINT_REG6	

The terminfo area is used for communication between CICS and Cobol II during RunUnit Termination.

(17C)	CHARACTER	236	TERMINFO	
(17C)	CHARACTER	4	TERMCODE	
(17C)	BITSTRING	1	*	
	1...		TERMCODE_BIT0	abnormal termination
	.1..		TERMCODE_BIT1	normal termination driven via EXEC CICS RETURN
	..1.		TERMCODE_BIT2	normal termination driven via native language return
	...1		TERMCODE_BIT3	normal termination driven in a called assembler rtn
	.... 1...		TERMCODE_BIT4	abend - ASRA
	.... .1..		TERMCODE_BIT5	abend - but not ASRA
	.... ..1.		TERMCODE_BIT6	lower level run-unit terminated abnormally
	1		TERMCODE_BIT7	user handle abend active
(17D)	BITSTRING	1	*	
	1...		TERMCODE_BIT8	This PTB in use
	.1..		TERMCODE_BIT9	interrupt in CICS
	..1.		TERMCODE_ BIT10	CICS dump suppressed
	...1		TERMCODE_ BIT11	abend_cancel active
	 1111		*	reserved
(17E)	BITSTRING	2	*	reserved
(180)	CHARACTER	4	ABCODE	
(184)	CHARACTER	8	PROGRAM_ CHECK_PSW	
(184)	CHARACTER	4	*	
(188)	CHARACTER	4	PROGRAM_CHECK_ADDRESS	
(18C)	CHARACTER	8	PROGRAM_CHECK_INTERRUPT_DATA	
(194)	CHARACTER	64	REGISTERS_AT_PROGRAM_CHECK	
(1D4)	CHARACTER	64	REGISTERS_AT_LAST_CICS_CMD	
(214)	FULLWORD	4	COBOL2_CONTCODE	
(218)	FULLWORD	4	RETRY_REGISTERS (16)	
(258)	CHARACTER	16	RETRY_PSW	

The celinfo area is used for communication between CICS and either C/370 during RunUnit Termination or LE/370 during RunUnit End Invocation.

(268)	CHARACTER	64	CELINFO	
(268)	CHARACTER	24	CELINFO_HEAD	
(268)	CHARACTER	4	*	
(26C)	CHARACTER	4	*	
(270)	CHARACTER	16	PSW	
(270)	CHARACTER	8	*	
(278)	CHARACTER	8	INTERRUPT_ DATA	
(278)	CHARACTER	2	INSTRUCTION_LENGTH	
(27A)	CHARACTER	2	INTERRUPT_ CODE	
(27C)	FULLWORD	4	EXCEPTION_ADDRESS	
(280)	ADDRESS	4	ABEND_GP_REGISTERS_ADDR	
(284)	ADDRESS	4	ABEND_FP_REGISTERS_ADDR	
(288)	ADDRESS	4	ABEND_AX_REGISTERS_ADDR	
(28C)	ADDRESS	4	LAST_CICS_CMD_REGISTERS_ ADDR	

The following area is completed by C/370 or LE/370.

(290)	CHARACTER	4	CONTCODE	
(290)	BITSTRING	1	*	
	1...		*	reserved
	.1..		CONTCODE_BIT1	retry using registers
	..1.		CONTCODE_BIT2	retry using PSW
	...1 1111		*	reserved
(291)	BITSTRING	3	*	reserved
(294)	CHARACTER	20	RETRY_DATA_ VECTOR	
(294)	FULLWORD	4	RETRY_ADDRESS	NB - there is no indirection
(298)	ADDRESS	4	RETRY_PROGRAM_MASK_ADDR	
(29C)	ADDRESS	4	RETRY_GP_REGISTERS_ADDR	
(2A0)	ADDRESS	4	RETRY_FP_REGISTERS_ADDR	
(2A4)	ADDRESS	4	RETRY_AX_REGISTERS_ADDR	

The language bits area is used during Determine Working Storage and Perform Goto calls to LE/370.

Offset Hex	Type	Len	Name (Dim)	Description
(2A8)	FULLWORD	4	LANGUAGE_BITS	
(2A8)	CHARACTER	1	BYTE1	
(2A9)	CHARACTER	3	*	
Special areas for decoding data returned by the Abend Manager.				
(2AC)	CHARACTER	4	TACB_ABEND_CODE	
(2B0)	CHARACTER	4	TACB_REG_ 13_AT_ABEND	

BAACT Bam process class

-

What follows defines the Business Application Manager Process class.

-

Protect against mulitple inclusion.

--
-

Changing these structure types will affect the format of the repository file records. Alter with care, and remember to consider the impacts on the Repository File Batch Utility - DFHBARUP.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	160	PROCESS	
INSTANCE DATA				
Declared Data				
(0)	CHAR Protected	153	INSTANCE_ DATA_BLOCK	
(0)	STRUCTURE IsA(EYE_CATCHER_TYPE) Protected	16	BAPR_EYE_ CATCHER	eye catcher
(0)	UNSIGNED Protected	2	EYE_LEN	object length
(2)	UNSIGNED Protected	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHAR Protected	12	EYE_STRING	'>DFHddxxxxxx'
(10)	SIGNED Protected	2	INSTANCE_ VERSION	
(12)	SIGNED Protected	2	INSTANCE_LENGTH	
(14)	ADDRESS Protected	4	TRANSIENT_PTR	->transient_state
(18)	CHAR Protected	8	PROTYPE_NAME	
(20)	STRUCTURE IsA(ACTIVITY_REF) Protected	56	ROOT_ACT_REF	
(20)	CHAR Public	50	ACT_KEY	Identification in dataset
(20)	CHAR Public	2	RTYPE	
(22)	CHAR Public	44	RID	
(22)	CHAR Public	44	*	
(22)	CHAR Public	44	PRO_ID	
(22)	CHAR Public	8	PTYPE_NAME	
(2A)	CHAR Public	36	PRO_NAME	
(22)	CHAR Public	44	REL_ACT_ID	
(22)	CHAR Public	27	UNIQUE_ID	like a Network UOWid
(22)	UNSIGNED Public	1	UID_LEN	
(23)	UNSIGNED Public	1	UID_LU_ LEN	
(24)	CHAR Public	25	*	
(3D)	CHAR Public	16	ACT_NAME	
(4D)	CHAR Public	1	*	
(4E)	FIXED Private	4	*	
(52)	CHAR Public	2	*	
(54)	ADDRESS Public	4	ACT_ADD	
(58)	OBJECT IsA(CONTAINER_SET) Protected	56	CONTAINERS	process containers

Offset Hex	Type	Len	Name (Dim)	Description

--
-

An instance of the Container_Set class consists of...

- items - number of container in the chain,

- size - size of buffer needed to flatten the container chain into,

- offset - in the flattened record this is the offset from this field to the container chain,

- chain - anchor for the container chain.

Offset Hex	Type	Len	Name (Dim)	Description
(58)	CHAR Protected	56	INSTANCE_DATA_BLOCK	
(58)	SIGNED Protected	4	ITEMS	
(5C)	SIGNED Protected	4	SIZE	
(60)	SIGNED Protected	4	CS_OFFSET	
(64)	CHAR Protected	4	*	
(68)	CHAR Protected	40	CHAIN	
Inherited Data				
(68)	CHAR Private	4	*	
(70)	CHAR Protected	16	ITER0	
(70)	CHAR Private	4	*	
(78)	CHAR Protected	8	*	
(78)	ADDRESS Protected	4	PREV	
(7C)	ADDRESS Protected	4	NEXT	
(80)	CHAR Protected	16	NODE0	
(80)	CHAR Private	4	*	
(88)	CHAR Protected	8	*	
(88)	ADDRESS Protected	4	PREV	
(8C)	ADDRESS Protected	4	NEXT	
(90)	FIXED Protected	1	AUDIT_LEVEL	Audit level
(91)	CHAR Protected	8	AUDIT_LOG	Audit log
(0)	STRUCTURE IsA(BAPR_TRANSIENT_STATE_TYPE) Protected	124	TRANSIENT_STATE	
(0)	ADDRESS Protected	4	PERMANENT_PTR	address of permanent state block
(4)	BITSTRING Protected	1	TRANSIENT_FLAGS	
	1... .... Protected		PR_READONLY	no write access
	.1.. .... Protected		UNFLATTENED	
	..1. .... Protected		*	
	...1 .... Protected		PRO_INSTORE	
(5)	CHAR Protected	3	*	
(8)	CHAR Protected	112	PROCESS_RECORD	Buffer

-

An instance of the buffer class contains the first in a list of segments. Segments are chained together if there is more data than can fit in one segment.

Offset Hex	Type	Len	Name (Dim)	Description
(8)	CHAR Public	112	INSTANCE_DATA_BLOCK	
(8)	CHAR Public	60	BABU_PUBLIC	
(8)	CHAR Public	8	FILENAME	file name
(10)	CHAR Public	50	KEY	key of object
(10)	CHAR Public	2	RTYPE	
(12)	CHAR Public	44	RID	
(12)	CHAR Public	44	*	
(12)	CHAR Public	44	PRO_ID	
(12)	CHAR Public	8	PTYPE_NAME	
(1A)	CHAR Public	36	PRO_NAME	
(12)	CHAR Public	44	REL_ACT_ ID	
(12)	CHAR Public	27	UNIQUE_ ID	like a Network UOWid
(12)	UNSIGNED Public	1	UID_LEN	
(13)	UNSIGNED Public	1	UID_LU_ LEN	
(14)	CHAR Public	25	*	
(2D)	CHAR Public	16	ACT_NAME	
(3D)	CHAR Public	1	*	

Offset Hex	Type	Len	Name (Dim)	Description
(3E)	FIXED Private	4	*	
(42)	CHAR Public	2	*	
(44)	CHAR Private	52	BABU_PRIVATE	
(44)	FIXED Private	1	BABU_BUF_ MODE	
(45)	FIXED Private	1	BABU_BUF_ STATE	
(46)	CHAR Private	2	*	
(48)	SIGNED Private	4	BABU_SEG_LEN	
(4C)	ADDRESS Private	4	BABU_SEG_LIST_HEAD	
(50)	ADDRESS Private	4	BABU_SEG_ LIST_TAIL	
(54)	ADDRESS Private	4	BABU_CURRENT_ PTR	
(58)	SIGNED Private	4	BABU_CURRENT_OFFS	
(5C)	STRUCTURE IsA(BABU_SEGMENT) Private	24	BABU_FIRST_ SEG	
(5C)	ADDRESS Protected	4	BABU_NEXT_ SEG	address of next segment
(60)	ADDRESS Protected	4	BABU_STG_ ADD	address of contents of segment
(64)	SIGNED Protected	4	BABU_STG_ LEN	length of storage in segment
(68)	SIGNED Protected	4	BABU_REC_ LEN	length of data in segment
(6C)	SIGNED Protected	4	BABU_SEQ	segment number
(70)	SIGNED Protected	4	BABU_FC_ UTOKEN	FC update token for segment
(74)	ADDRESS Private	4	BABU_WRITE_STG_ADD	
(78)	CHAR Protected	4	SOURCE_REF	
(78)	ADDRESS Protected	4	ACT_REQ_PTR	

SHARED DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Public	56	NULL_PRO_REF	
(0)	CHAR Public	56	PROCESS_REF	
(0)	STRUCTURE IsA(BALR_KEY) Public	50	PRO_KEY	
(0)	CHAR Public	2	RTYPE	
(2)	CHAR Public	44	RID	
(2)	CHAR Public	44	*	
(2)	CHAR Public	44	PRO_ID	
(2)	CHAR Public	8	PTYPE_NAME	
(A)	CHAR Public	36	PRO_NAME	
(2)	CHAR Public	44	REL_ACT_ID	
(2)	CHAR Public	27	UNIQUE_ID	like a Network UOWid
(2)	UNSIGNED Public	1	UID_LEN	
(3)	UNSIGNED Public	1	UID_LU_LEN	
(4)	CHAR Public	25	*	
(1D)	CHAR Public	16	ACT_NAME	
(2D)	CHAR Public	1	*	
(2E)	FIXED Private	4	*	
(32)	CHAR Public	2	*	
(34)	ADDRESS Public	4	PRO_ADD	
(0)	CHAR Protected	124	BAPR_TRANSIENT_STATE_TYPE	
(0)	ADDRESS Protected	4	PERMANENT_PTR	address of permanent state block
(4)	BITSTRING Protected	1	TRANSIENT_FLAGS	
	1... .... Protected		PR_READONLY	no write access
	.1.. .... Protected		UNFLATTENED	
	..1. .... Protected		*	
	...1 .... Protected		PRO_INSTORE	
(5)	CHAR Protected	3	*	
(8)	OBJECT IsA(BABU) Protected	112	PROCESS_RECORD	Buffer
(8)	CHAR Public	112	INSTANCE_DATA_BLOCK	
(8)	CHAR Public	60	BABU_PUBLIC	
(8)	CHAR Public	8	FILENAME	file name
(10)	CHAR Public	50	KEY	key of object
(10)	CHAR Public	2	RTYPE	
(12)	CHAR Public	44	RID	
(12)	CHAR Public	44	*	
(12)	CHAR Public	44	PRO_ID	
(12)	CHAR Public	8	PTYPE_NAME	

Offset Hex	Type	Len	Name (Dim)	Description
(1A)	CHAR Public	36	PRO_NAME	
(12)	CHAR Public	44	REL_ACT_ ID	
(12)	CHAR Public	27	UNIQUE_ ID	like a Network UOWid
(12)	UNSIGNED Public	1	UID_LEN	
(13)	UNSIGNED Public	1	UID_LU_ LEN	
(14)	CHAR Public	25	*	
(2D)	CHAR Public	16	ACT_NAME	
(3D)	CHAR Public	1	*	
(3E)	FIXED Private	4	*	Buffer
(42)	CHAR Public	2	*	Buffer
(44)	CHAR Private	52	BABU_PRIVATE	Buffer
(44)	FIXED Private	1	BABU_BUF_ MODE	Buffer
(45)	FIXED Private	1	BABU_BUF_ STATE	Buffer
(46)	CHAR Private	2	*	Buffer
(48)	SIGNED Private	4	BABU_SEG_LEN	Buffer
(4C)	ADDRESS Private	4	BABU_SEG_ LIST_HEAD	Buffer
(50)	ADDRESS Private	4	BABU_SEG_ LIST_TAIL	Buffer
(54)	ADDRESS Private	4	BABU_CURRENT_ PTR	Buffer
(58)	SIGNED Private	4	BABU_CURRENT_ OFFS	Buffer
(5C)	STRUCTURE IsA(BABU_SEGMENT) Private	24	BABU_FIRST_ SEG	Buffer
(5C)	ADDRESS Protected	4	BABU_NEXT_ SEG	address of next segment
(60)	ADDRESS Protected	4	BABU_STG_ ADD	address of contents of segment
(64)	SIGNED Protected	4	BABU_STG_ LEN	length of storage in segment
(68)	SIGNED Protected	4	BABU_REC_ LEN	length of data in segment
(6C)	SIGNED Protected	4	BABU_SEQ	segment number
(70)	SIGNED Protected	4	BABU_FC_ UTOKEN	FC update token for segment
(74)	ADDRESS Private	4	BABU_WRITE_ STG_ADD	Buffer
(78)	CHAR Protected	4	SOURCE_REF	
(78)	ADDRESS Protected	4	ACT_REQ_PTR	pro_instore - act request

Constants

Len	Type	Value	Name	Description
2	CHARACTER	P	BAPR_PROCESS_ RECORD_TYPE	
2	DECIMAL	1	BAPR_PROCESS_ INSTANCE_VER_1	

The length occupied by a Process object in a repository record is currently set as 200 bytes. This leaves some space should the data in the flat form of the object need to increase.

Len	Type	Value	Name	Description
4	DECIMAL	200	FLAT_PROCESS_ LENGTH	

A dummy based variable is declared to provide a compile time check that the flat length is sufficient to accomodate the real object.

Len	Type	Value	Name	Description
4	DECIMAL	40	FLAT_PROCESS_SPARE	

BAACT Bam contaier_set class

-

What follows defines the Business Application Manager Container_Set class.

-

Protect against multiple inclusion.

--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	56	CONTAINER_SET	

--
-

An instance of the Container_ Set class consists of...

- items - number of container in the chain,

- size - size of buffer needed to flatten the container chain into,

- offset - in the flattened record this is the offset from this field to the container chain,

- chain - anchor for the container chain.

INSTANCE DATA				
Declared Data				
(0)	CHAR Protected	56	INSTANCE_ DATA_BLOCK	
(0)	SIGNED Protected	4	ITEMS	
(4)	SIGNED Protected	4	SIZE	
(8)	SIGNED Protected	4	CS_OFFSET	
(C)	CHAR Protected	4	*	padding
(10)	OBJECT IsA(HOP_DCHAIN) Protected	40	CHAIN	
Inherited Data				
(10)	CHAR Private	4	*	
(18)	CHAR Protected	16	ITER0	
(18)	CHAR Private	4	*	
(20)	CHAR Protected	8	*	
(20)	ADDRESS Protected	4	PREV	
(24)	ADDRESS Protected	4	NEXT	
(28)	CHAR Protected	16	NODE0	
(28)	CHAR Private	4	*	
(30)	CHAR Protected	8	*	
(30)	ADDRESS Protected	4	PREV	
(34)	ADDRESS Protected	4	NEXT	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	12	BACS_CONTAINER_NOT_FOUND	
4	DECIMAL	11	BACS_LENGTH_ERROR	
4	DECIMAL	24	BACS_INVALID_CONTAINER_NAME	
1	HEX	FF	HOP_TRUE	
1	HEX	00	HOP_FALSE	

BAACT Bam activity class

What follows defines the Business Application Manager Event
Driven Object Class.
Protect against mulitple inclusion.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	336	ACTIVITY	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Protected	20	BAEV_INSTANCE_DATA_BLOCK	
(0)	CHAR Protected	16	BAEV_EYE_ CATCHER	eye catcher
(0)	UNSIGNED Protected	2	EYE_LEN	object length
(2)	UNSIGNED Protected	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHAR Protected	12	EYE_STRING	'>DFHddxxxxxx'
(10)	SIGNED Protected	4	EVENT_POOL_ TOKEN	event pool token

--
 -

An instance of the Activity class consists of...

Offset Hex	Type	Len	Name (Dim)	Description
Declared Data				
(18)	CHAR Protected	306	INSTANCE_ DATA_BLOCK	
(18)	SIGNED Protected	2	INSTANCE_LENGTH	
(1A)	SIGNED Protected	2	INSTANCE_ VERSION	
(1C)	ADDRESS Protected	4	TRANSIENT_PTR	@ transient_state
(20)	STRUCTURE IsA(BAAC_PERMANENT_STATE_TYPE) Protected	298	PERMANENT_STATE	
(20)	CHAR Protected	50	OWN_PROCESS	owning process
(20)	CHAR Public	2	RTYPE	
(22)	CHAR Public	44	RID	
(22)	CHAR Public	44	*	
(22)	CHAR Public	44	PRO_ID	
(22)	CHAR Public	8	PTYPE_NAME	
(2A)	CHAR Public	36	PRO_NAME	
(22)	CHAR Public	44	REL_ACT_ID	
(22)	CHAR Public	27	UNIQUE_ID	like a Network UOWid
(22)	UNSIGNED Public	1	UID_LEN	
(23)	UNSIGNED Public	1	UID_LU_ LEN	
(24)	CHAR Public	25	*	
(3D)	CHAR Public	16	ACT_NAME	
(4D)	CHAR Public	1	*	
(4E)	FIXED Private	4	*	
(52)	STRUCTURE IsA(BALR_KEY) Protected	50	PARENT_KEY	
(52)	CHAR Public	2	RTYPE	
(54)	CHAR Public	44	RID	
(54)	CHAR Public	44	*	
(54)	CHAR Public	44	PRO_ID	
(54)	CHAR Public	8	PTYPE_NAME	
(5C)	CHAR Public	36	PRO_NAME	
(54)	CHAR Public	44	REL_ACT_ID	
(54)	CHAR Public	27	UNIQUE_ID	like a Network UOWid

Offset Hex	Type	Len	Name (Dim)	Description
(54)	UNSIGNED Public	1	UID_LEN	
(55)	UNSIGNED Public	1	UID_LU_ LEN	
(56)	CHAR Public	25	*	
(6F)	CHAR Public	16	ACT_NAME	
(7F)	CHAR Public	1	*	
(80)	FIXED Private	4	*	
(84)	CHAR Protected	27	OWN_ROOT_ID	
(9F)	FIXED Protected	1	MODE	
(A0)	CHAR Protected	4	PARENT_TRANID	
(A4)	CHAR Protected	8	PARENT_USERID	
(AC)	UNSIGNED Protected	1	STARTED	
(AD)	UNSIGNED Protected	1	BLOCKED	
(AE)	CHAR Protected	2	*	
(B0)	SIGNED Protected	4	PARENT_ GENERATION	parent gen_num
(B4)	CHAR Protected	8	CHILDREN	
(B4)	UNSIGNED Protected	4	N	number of activities
(B8)	ADDRESS Protected	4	HEAD	head of list of activities
(BC)	ADDRESS Protected	4	FLAT_EPOOL_PTR	Flat EM state address
(C0)	SIGNED Protected	4	FLAT_EPOOL_LEN	Flat EM state length
(C4)	SIGNED Protected	4	GENERATION	Generation Number
(C8)	CHAR Protected	56	CONTAINERS	

--
-

An instance of the Container_ Set class consists of...

- items - number of container in the chain,

- size - size of buffer needed to flatten the container chain into,

- offset - in the flattened record this is the offset from this field to the container chain,

- chain - anchor for the container chain.

(C8)	CHAR Protected	56	INSTANCE_ DATA_BLOCK	
(C8)	SIGNED Protected	4	ITEMS	
(CC)	SIGNED Protected	4	SIZE	
(D0)	SIGNED Protected	4	CS_OFFSET	
(D4)	CHAR Protected	4	*	
(D8)	CHAR Protected	40	CHAIN	
(D8)	CHAR Private	4	*	
(E0)	CHAR Protected	16	ITER0	
(E0)	CHAR Private	4	*	
(E8)	CHAR Protected	8	*	
(E8)	ADDRESS Protected	4	PREV	
(EC)	ADDRESS Protected	4	NEXT	
(F0)	CHAR Protected	16	NODE0	
(F0)	CHAR Private	4	*	
(F8)	CHAR Protected	8	*	
(F8)	ADDRESS Protected	4	PREV	
(FC)	ADDRESS Protected	4	NEXT	
(100)	CHAR Protected	44	ATTRIBUTES	
(100)	CHAR Protected	8	PROGRAM	program name
(108)	CHAR Protected	8	*	
(110)	CHAR Protected	4	TRANID	transaction ID
(114)	CHAR Protected	8	USERID	user identifier
(11C)	CHAR Protected	16	COMPLETION_ EVENT	completion event
(12C)	CHAR Protected	13	COMPLETION_ DATA	
(12C)	UNSIGNED Public	1	COMPLETION_ RESP	
(12D)	CHAR Public	4	AB_CODE	
(131)	CHAR Public	8	AB_PROGRAM	

Offset Hex	Type	Len	Name (Dim)	Description
(139)	UNSIGNED Protected	1	AUDIT_LEVEL	Audit level
(13A)	CHAR Protected	8	AUDIT_LOG	Audit log name
(142)	CHAR Protected	8	*	

--

(0)	CHAR Protected	8	PTYPE	
(0)	CHAR Protected	36	PNAME	
(0)	STRUCTURE IsA(BAAC_TRANSIENT_STATE_TYPE) Protected	136	TRANSIENT_STATE	
(0)	BITSTRING Protected	1	TRANSIENT_FLAGS	
	1... .... Protected		ACT_INSTORE	
	.1.. .... Protected		ACT_IN_BUFFERS	
	..1. .... Protected		*	
	...1 .... Protected		*	
	.... 1... Protected		*	
	.... .1.. Protected		ACTIVATED	
	.... ..1. Protected		RET_ENDACTIVITY	EndActivity specified on return
	....1 Protected		*	
(1)	CHAR Protected	3	*	
(4)	CHAR Protected	112	ACTIVITY_RECORD	buffers for record data

-

An instance of the buffer class contains the first in a list of segments. Segments are chained together if there is more data than can fit in one segment.

(4)	CHAR Public	112	INSTANCE_ DATA_BLOCK	
(4)	CHAR Public	60	BABU_PUBLIC	
(4)	CHAR Public	8	FILENAME	file name
(C)	CHAR Public	50	KEY	key of object
(C)	CHAR Public	2	RTYPE	
(E)	CHAR Public	44	RID	
(E)	CHAR Public	44	*	
(E)	CHAR Public	44	PRO_ID	
(E)	CHAR Public	8	PTYPE_NAME	
(16)	CHAR Public	36	PRO_NAME	
(E)	CHAR Public	44	REL_ACT_ ID	
(E)	CHAR Public	27	UNIQUE_ ID	like a Network UOWid
(E)	UNSIGNED Public	1	UID_LEN	
(F)	UNSIGNED Public	1	UID_LU_ LEN	
(10)	CHAR Public	25	*	
(29)	CHAR Public	16	ACT_NAME	
(39)	CHAR Public	1	*	
(3A)	FIXED Private	4	*	
(3E)	CHAR Public	2	*	
(40)	CHAR Private	52	BABU_PRIVATE	
(40)	FIXED Private	1	BABU_BUF_ MODE	
(41)	FIXED Private	1	BABU_BUF_ STATE	
(42)	CHAR Private	2	*	
(44)	SIGNED Private	4	BABU_SEG_LEN	
(48)	ADDRESS Private	4	BABU_SEG_ LIST_HEAD	
(4C)	ADDRESS Private	4	BABU_SEG_ LIST_TAIL	
(50)	ADDRESS Private	4	BABU_CURRENT_ PTR	
(54)	SIGNED Private	4	BABU_CURRENT_ OFFS	
(58)	STRUCTURE IsA(BABU_SEGMENT) Private	24	BABU_FIRST_ SEG	
(58)	ADDRESS Protected	4	BABU_NEXT_ SEG	address of next segment
(5C)	ADDRESS Protected	4	BABU_STG_ ADD	address of contents of segment
(60)	SIGNED Protected	4	BABU_STG_ LEN	length of storage in segment
(64)	SIGNED Protected	4	BABU_REC_ LEN	length of data in segment
(68)	SIGNED Protected	4	BABU_SEQ	segment number
(6C)	SIGNED Protected	4	BABU_FC_ UTOKEN	
				FC update token for segment
(70)	ADDRESS Private	4	BABU_WRITE_ STG_ADD	
(74)	ADDRESS Protected	4	PERMANENT_PTR	

Offset Hex	Type	Len	Name (Dim)	Description
(78)	CHAR Protected	4	SOURCE_REF	
(78)	ADDRESS Protected	4	ACT_REQ_PTR	
(7C)	ADDRESS Protected	4	PARENT_ADD	
(80)	ADDRESS Protected	4	NEXT	
(84)	ADDRESS Protected	4	PREV	

-

Changing these structure types will affect the format of the repository file records. Alter with care, and remember to consider the impacts on the Repository File Batch Utility - DFHBARUP.

-

This is a very important type within the Activity Class.

For an activity, it associates a parental activity name (how the activity program of a parent refers to a child activity), with the token to the activity state in the dataset (Repository File) and any in-memory instantiation of the activity that might exist.

Each activity may contain many instances of this type.

relative_ activity_id
how the activity is identified in the dataset
act_add
address of start of this activity object

parent
identification of this activity's parent
children
identification of child activities in the child_ set.

SHARED DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Public	56	ACTIVITY_REF	
(0)	STRUCTURE IsA(BALR_KEY) Public	50	ACT_KEY	Identification in dataset
(0)	CHAR Public	2	RTYPE	
(2)	CHAR Public	44	RID	
(2)	CHAR Public	44	*	
(2)	CHAR Public	44	PRO_ID	
(2)	CHAR Public	8	PTYPE_NAME	
(A)	CHAR Public	36	PRO_NAME	
(2)	CHAR Public	44	REL_ACT_ID	
(2)	CHAR Public	27	UNIQUE_ID	like a Network UOWid
(2)	UNSIGNED Public	1	UID_LEN	
(3)	UNSIGNED Public	1	UID_LU_LEN	
(4)	CHAR Public	25	*	
(1D)	CHAR Public	16	ACT_NAME	
(2D)	CHAR Public	1	*	
(2E)	FIXED Private	4	*	Identification in dataset
(32)	CHAR Public	2	*	
(34)	ADDRESS Public	4	ACT_ADD	Instantiated object address

-

Here are the various definitional attributes of activities.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	44	ACTIVITY_ATTRIBS	
(0)	CHAR Protected	8	PROGRAM	program name
(8)	CHAR Protected	8	*	reserved
(10)	CHAR Protected	4	TRANID	transaction ID
(14)	CHAR Protected	8	USERID	user identifier
(1C)	CHAR Protected	16	COMPLETION_ EVENT	completion event

-

Here are various attributes of the activity relevant at completion.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Public	13	ACTIVITY_ COMP_DATA	
(0)	FIXED Public	1	COMPLETION_RESP	
(1)	CHAR Public	4	AB_CODE	
(5)	CHAR Public	8	AB_PROGRAM	
(0)	CHAR Protected	8	ACTIVITY_SET	
(0)	UNSIGNED Protected	4	N	number of activities

Offset Hex	Type	Len	Name (Dim)	Description
(4)	ADDRESS Protected	4	HEAD	head of list of activities

Every member in a activity_set contains an activity_ref to the activity and some element attributes.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	FIXED Protected	4	CHILD_MODE	
(0)	CHAR Protected	69	ACTIVITY_ SET_ELEMENT	
(0)	ADDRESS Protected	4	NEXT_ELEM	pointer to next in set
(4)	STRUCTURE IsA(ACTIVITY_REF) Protected	56	ACT_REF	identification of activity
(4)	CHAR Public	50	ACT_KEY	Identification in dataset
(4)	CHAR Public	2	RTYPE	
(6)	CHAR Public	44	RID	
(6)	CHAR Public	44	*	
(6)	CHAR Public	44	PRO_ID	
(6)	CHAR Public	8	PTYPE_NAME	
(E)	CHAR Public	36	PRO_NAME	
(6)	CHAR Public	44	REL_ACT_ID	
(6)	CHAR Public	27	UNIQUE_ID	like a Network UOWid
(6)	UNSIGNED Public	1	UID_LEN	
(7)	UNSIGNED Public	1	UID_LU_ LEN	
(8)	CHAR Public	25	*	
(21)	CHAR Public	16	ACT_NAME	
(31)	CHAR Public	1	*	
(32)	FIXED Private	4	*	
(36)	CHAR Public	2	*	
(38)	ADDRESS Public	4	ACT_ADD	identification of activity
(3C)	SIGNED Protected	4	SUB_GEN_NO	generation no of child
(40)	FIXED Protected	4	SUB_MODE	simplified mode of child
(44)	BITSTRING Protected	1	*	
	1... .... Protected		UNFLATTENED	
	.111 1111 Protected		*	
(0)	CHAR Protected	11	FLAT_SET_ ELEMENT_SPACE	

--
--
-

These are the modes of the activity, as documented in the specificaion DFHBAZED.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	FIXED Public	1	ACT_MODE	

--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	FIXED Public	1	ACT_COMPLETION_ RESP	

-

This is a fully qualified identification of the activity, used in Scheduler Services requests. It includes the generation number of the activity.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Public	112	ACTIVITY_ID	
(0)	STRUCTURE IsA(BARF) Public	8	PROC_FILE	
(8)	STRUCTURE IsA(BALR_KEY) Public	50	PRO_LR_KEY	
(8)	CHAR Public	2	RTYPE	
(A)	CHAR Public	44	RID	
(A)	CHAR Public	44	*	
(A)	CHAR Public	44	PRO_ID	
(A)	CHAR Public	8	PTYPE_NAME	
(12)	CHAR Public	36	PRO_NAME	
(A)	CHAR Public	44	REL_ACT_ID	
(A)	CHAR Public	27	UNIQUE_ID	like a Network UOWid
(A)	UNSIGNED Public	1	UID_LEN	
(B)	UNSIGNED Public	1	UID_LU_LEN	
(C)	CHAR Public	25	*	
(25)	CHAR Public	16	ACT_NAME	
(35)	CHAR Public	1	*	

Offset Hex	Type	Len	Name (Dim)	Description
(36)	FIXED Private	4	*	
(3A)	STRUCTURE IsA(BALR_KEY) Public	50	ACT_LR_KEY	
(3A)	CHAR Public	2	RTYPE	
(3C)	CHAR Public	44	RID	
(3C)	CHAR Public	44	*	
(3C)	CHAR Public	44	PRO_ID	
(3C)	CHAR Public	8	PTYPE_NAME	
(44)	CHAR Public	36	PRO_NAME	
(3C)	CHAR Public	44	REL_ACT_ID	
(3C)	CHAR Public	27	UNIQUE_ID	like a Network UOWid
(3C)	UNSIGNED Public	1	UID_LEN	
(3D)	UNSIGNED Public	1	UID_LU_LEN	
(3E)	CHAR Public	25	*	
(57)	CHAR Public	16	ACT_NAME	
(67)	CHAR Public	1	*	
(68)	FIXED Private	4	*	
(6C)	SIGNED Public	4	ACT_GEN_NO	

--
-

A request, passed on SH (but encapsulated) and passed to BAXM and field types and constants.

Request_ Action the basic type of request being made

(0)	FIXED Public	1	REQUEST_ACTION

Request_ Reason the reason for the request (varies with action)

(0)	FIXED Public	1	REQUEST_REASON	
(0)	CHAR Public	275	ACTIVITY_REQUEST	
(0)	STRUCTURE IsA(ACTIVITY_ID) Public	112	TARGET	
(0)	CHAR Public	8	PROC_FILE	
(8)	CHAR Public	50	PRO_LR_KEY	
(8)	CHAR Public	2	RTYPE	
(A)	CHAR Public	44	RID	
(A)	CHAR Public	44	*	
(A)	CHAR Public	44	PRO_ID	
(A)	CHAR Public	8	PTYPE_NAME	
(12)	CHAR Public	36	PRO_NAME	
(A)	CHAR Public	44	REL_ACT_ID	
(A)	CHAR Public	27	UNIQUE_ID	like a Network UOWid
(A)	UNSIGNED Public	1	UID_LEN	
(B)	UNSIGNED Public	1	UID_LU_ LEN	
(C)	CHAR Public	25	*	
(25)	CHAR Public	16	ACT_NAME	
(35)	CHAR Public	1	*	
(36)	FIXED Private	4	*	
(3A)	STRUCTURE IsA(BALR_KEY) Public	50	ACT_LR_KEY	
(3A)	CHAR Public	2	RTYPE	
(3C)	CHAR Public	44	RID	
(3C)	CHAR Public	44	*	
(3C)	CHAR Public	44	PRO_ID	
(3C)	CHAR Public	8	PTYPE_NAME	
(44)	CHAR Public	36	PRO_NAME	
(3C)	CHAR Public	44	REL_ACT_ID	
(3C)	CHAR Public	27	UNIQUE_ID	like a Network UOWid
(3C)	UNSIGNED Public	1	UID_LEN	
(3D)	UNSIGNED Public	1	UID_LU_ LEN	
(3E)	CHAR Public	25	*	
(57)	CHAR Public	16	ACT_NAME	
(67)	CHAR Public	1	*	
(68)	FIXED Private	4	*	
(6C)	SIGNED Public	4	ACT_GEN_NO	
(70)	STRUCTURE IsA(ACTIVITY_ID) Public	112	ORIGIN	
(70)	CHAR Public	8	PROC_FILE	
(78)	CHAR Public	50	PRO_LR_KEY	
(78)	CHAR Public	2	RTYPE	
(7A)	CHAR Public	44	RID	
(7A)	CHAR Public	44	*	
(7A)	CHAR Public	44	PRO_ID	
(7A)	CHAR Public	8	PTYPE_NAME	
(82)	CHAR Public	36	PRO_NAME	

Offset Hex	Type	Len	Name (Dim)	Description
(7A)	CHAR Public	44	REL_ACT_ID	
(7A)	CHAR Public	27	UNIQUE_ID	like a Network UOWid
(7A)	UNSIGNED Public	1	UID_LEN	
(7B)	UNSIGNED Public	1	UID_LU_ LEN	
(7C)	CHAR Public	25	*	
(95)	CHAR Public	16	ACT_NAME	
(A5)	CHAR Public	1	*	
(A6)	FIXED Private	4	*	
(AA)	STRUCTURE IsA(BALR_KEY) Public	50	ACT_LR_KEY	
(AA)	CHAR Public	2	RTYPE	
(AC)	CHAR Public	44	RID	
(AC)	CHAR Public	44	*	
(AC)	CHAR Public	44	PRO_ID	
(AC)	CHAR Public	8	PTYPE_NAME	
(B4)	CHAR Public	36	PRO_NAME	
(AC)	CHAR Public	44	REL_ACT_ID	
(AC)	CHAR Public	27	UNIQUE_ID	like a Network UOWid
(AC)	UNSIGNED Public	1	UID_LEN	
(AD)	UNSIGNED Public	1	UID_LU_ LEN	
(AE)	CHAR Public	25	*	
(C7)	CHAR Public	16	ACT_NAME	
(D7)	CHAR Public	1	*	
(D8)	FIXED Private	4	*	
(DC)	SIGNED Public	4	ACT_GEN_NO	
(E0)	STRUCTURE IsA(IN_STORE_TARGET) Public	16	IS_TARGET	iff in_store='1'b
(E0)	ADDRESS Public	4	IS_ACT_PTR	
(E4)	SIGNED Public	4	IS_ACT_LEN	
(E8)	ADDRESS Public	4	IS_PRO_PTR	
(EC)	SIGNED Public	4	IS_PRO_LEN	
(F0)	CHAR Public	16	EVENT	fire parm
(100)	UNSIGNED Public	4	EVENT_VERSION	event version (or zero)
(104)	BITSTRING Public	1	REQUEST_FLAGS	
	1... .... Protected		IN_STORE	
	.1.. .... Protected		BAD_EVENT	
	..1. .... Protected		BRIDGE_X	
	...1 1111 Protected		*	
(105)	FIXED Public	1	REQ_TYPE	
(106)	FIXED Public	1	REQ_REASON	why request
(107)	CHAR Public	4	ORIGIN_TRANID	
(10B)	CHAR Public	8	BRIDGE_ FACILITY_TOKEN	
(0)	CHAR Public	16	IN_STORE_TARGET	
(0)	ADDRESS Public	4	IS_ACT_PTR	
(4)	SIGNED Public	4	IS_ACT_LEN	
(8)	ADDRESS Public	4	IS_PRO_PTR	
(C)	SIGNED Public	4	IS_PRO_LEN	

--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	FIXED Public	1	EXEC_MODE	

--
-

Class Data for the Activity Class is declared as a private type. Storage for it is obtained for a single instance of the type from BADM during initialisation. BADM also looks after addressing it (via badm_ set/inq_ class_data).

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	88	BAAC_CLASS_ DATA_TYPE	
(0)	STRUCTURE IsA(EYE_CATCHER_TYPE) Protected	16	CLASS_EYE_ CATCHER	eye catcher
(0)	UNSIGNED Protected	2	EYE_LEN	object length
(2)	UNSIGNED Protected	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHAR Protected	12	EYE_STRING	'>DFHddxxxxxx'
(10)	OBJECT IsA(BAOF) Protected	40	TRANSIENT_ OBJECT_FACTORY	object factory for transient state

Offset Hex	Type	Len	Name (Dim)	Description
-				
The instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix 'BAOF' and a suffix which is the name of the object being managed.				
(10)	CHAR Protected	40	INSTANCE_ DATA_BLOCK	BAOF instance data
(10)	CHAR Protected	16	OF_EYE_ CATCHER	eye-catcher
(10)	UNSIGNED Protected	2	EYE_LEN	object length
(12)	UNSIGNED Protected	2	EYE_OFFSET	offset of eye-catcher in object
(14)	CHAR Protected	12	EYE_STRING	'>DFHddxxxxxx'
(20)	CHAR Protected	8	SUBPOOL_NAME	subpool name
(20)	CHAR Protected	4	SUBPOOL_ NAME_PREFIX	subpool name prefix
(24)	CHAR Protected	4	SUBPOOL_ NAME_SUFFIX	subpool name suffix
(28)	CHAR Protected	8	SUBPOOL_TOKEN	subpool token
(30)	CHAR Protected	8	*	
(38)	CHAR Protected	32	*	
(0)	CHAR Protected	298	BAAC_PERMANENT_ STATE_TYPE	
(0)	STRUCTURE IsA(BALR_KEY) Protected	50	OWN_PROCESS	owning process
(0)	CHAR Public	2	RTYPE	
(2)	CHAR Public	44	RID	
(2)	CHAR Public	44	*	
(2)	CHAR Public	44	PRO_ID	
(2)	CHAR Public	8	PTYPE_NAME	
(A)	CHAR Public	36	PRO_NAME	
(2)	CHAR Public	44	REL_ACT_ID	
(2)	CHAR Public	27	UNIQUE_ID	like a Network UOWid
(2)	UNSIGNED Public	1	UID_LEN	
(3)	UNSIGNED Public	1	UID_LU_LEN	
(4)	CHAR Public	25	*	
(1D)	CHAR Public	16	ACT_NAME	
(2D)	CHAR Public	1	*	
(2E)	FIXED Private	4	*	
(32)	STRUCTURE IsA(BALR_KEY) Protected	50	PARENT_KEY	parent Activity
(32)	CHAR Public	2	RTYPE	
(34)	CHAR Public	44	RID	
(34)	CHAR Public	44	*	
(34)	CHAR Public	44	PRO_ID	
(34)	CHAR Public	8	PTYPE_NAME	
(3C)	CHAR Public	36	PRO_NAME	
(34)	CHAR Public	44	REL_ACT_ID	
(34)	CHAR Public	27	UNIQUE_ID	like a Network UOWid
(34)	UNSIGNED Public	1	UID_LEN	
(35)	UNSIGNED Public	1	UID_LU_LEN	
(36)	CHAR Public	25	*	
(4F)	CHAR Public	16	ACT_NAME	
(5F)	CHAR Public	1	*	
(60)	FIXED Private	4	*	
(64)	CHAR Protected	27	OWN_ROOT_ID	root id
(7F)	FIXED Protected	1	MODE	this activity mode
(80)	CHAR Protected	4	PARENT_TRANID	
(84)	CHAR Protected	8	PARENT_USERID	
(8C)	UNSIGNED Protected	1	STARTED	
(8D)	UNSIGNED Protected	1	BLOCKED	
(8E)	CHAR Protected	2	*	
(90)	SIGNED Protected	4	PARENT_ GENERATION	parent gen_num
(94)	STRUCTURE IsA(ACTIVITY_SET) Protected	8	CHILDREN	
(94)	UNSIGNED Protected	4	N	number of activities
(98)	ADDRESS Protected	4	HEAD	head of list of activities

Offset Hex	Type	Len	Name (Dim)	Description
(9C)	ADDRESS Protected	4	FLAT_EPOOL_PTR	Flat EM state address
(A0)	SIGNED Protected	4	FLAT_EPOOL_LEN	Flat EM state length
(A4)	SIGNED Protected	4	GENERATION	Generation Number
(A8)	OBJECT IsA(CONTAINER_SET) Protected	56	CONTAINERS	
(A8)	CHAR Protected	56	INSTANCE_DATA_BLOCK	
(A8)	SIGNED Protected	4	ITEMS	
(AC)	SIGNED Protected	4	SIZE	
(B0)	SIGNED Protected	4	CS_OFFSET	
(B4)	CHAR Protected	4	*	
(B8)	CHAR Protected	40	CHAIN	
Inherited Data				
(B8)	CHAR Private	4	*	
(C0)	CHAR Protected	16	ITER0	
(C0)	CHAR Private	4	*	
(C8)	CHAR Protected	8	*	
(C8)	ADDRESS Protected	4	PREV	
(CC)	ADDRESS Protected	4	NEXT	
(D0)	CHAR Protected	16	NODE0	
(D0)	CHAR Private	4	*	
(D8)	CHAR Protected	8	*	
(D8)	ADDRESS Protected	4	PREV	
(DC)	ADDRESS Protected	4	NEXT	
(E0)	STRUCTURE IsA(ACTIVITY_ATTRIBS) Protected	44	ATTRIBUTES	
(E0)	CHAR Protected	8	PROGRAM	program name
(E8)	CHAR Protected	8	*	
(F0)	CHAR Protected	4	TRANID	transaction ID
(F4)	CHAR Protected	8	USERID	user identifier
(FC)	CHAR Protected	16	COMPLETION_ EVENT	completion event
(10C)	STRUCTURE IsA(ACTIVITY_COMP_DATA) Protected	13	COMPLETION_DATA	
(10C)	UNSIGNED Public	1	COMPLETION_ RESP	
(10D)	CHAR Public	4	AB_CODE	
(111)	CHAR Public	8	AB_PROGRAM	
(119)	FIXED Protected	1	AUDIT_LEVEL	Audit level
(11A)	CHAR Protected	8	AUDIT_LOG	Audit log name
(122)	CHAR Protected	8	*	
(0)	CHAR Protected	136	BAAC_TRANSIENT_STATE_TYPE	
(0)	BITSTRING Protected	1	TRANSIENT_FLAGS	
	`1... ....` Protected		ACT_INSTORE	
	`.1.. ....` Protected		ACT_IN_BUFFERS	
	`..1. ....` Protected		*	
	`...1 ....` Protected		*	
	`.... 1...` Protected		*	
	`.... .1..` Protected		ACTIVATED	
	`.... ..1.` Protected		RET_ENDACTIVITY	EndActivity specified on return
	`.... ...1` Protected		*	
(1)	CHAR Protected	3	*	
(4)	OBJECT IsA(BABU) Protected	112	ACTIVITY_RECORD	buffers for record data
(4)	CHAR Public	112	INSTANCE_DATA_BLOCK	
(4)	CHAR Public	60	BABU_PUBLIC	
(4)	CHAR Public	8	FILENAME	file name
(C)	CHAR Public	50	KEY	key of object
(C)	CHAR Public	2	RTYPE	
(E)	CHAR Public	44	RID	
(E)	CHAR Public	44	*	
(E)	CHAR Public	44	PRO_ID	
(E)	CHAR Public	8	PTYPE_NAME	
(16)	CHAR Public	36	PRO_NAME	
(E)	CHAR Public	44	REL_ACT_ ID	
(E)	CHAR Public	27	UNIQUE_ ID	like a Network UOWid
(E)	UNSIGNED Public	1	UID_LEN	

Offset Hex	Type	Len	Name (Dim)	Description
(F)	UNSIGNED Public	1	UID_LU_ LEN	
(10)	CHAR Public	25	*	
(29)	CHAR Public	16	ACT_NAME	
(39)	CHAR Public	1	*	
(3A)	FIXED Private	4	*	
(3E)	CHAR Public	2	*	
(40)	CHAR Private	52	BABU_PRIVATE	buffers for record data
(40)	FIXED Private	1	BABU_BUF_ MODE	buffers for record data
(41)	FIXED Private	1	BABU_BUF_ STATE	buffers for record data
(42)	CHAR Private	2	*	
(44)	SIGNED Private	4	BABU_SEG_LEN	buffers for record data
(48)	ADDRESS Private	4	BABU_SEG_ LIST_HEAD	buffers for record data
(4C)	ADDRESS Private	4	BABU_SEG_ LIST_TAIL	buffers for record data
(50)	ADDRESS Private	4	BABU_CURRENT_ PTR	buffers for record data
(54)	SIGNED Private	4	BABU_CURRENT_ OFFS	buffers for record data
(58)	STRUCTURE IsA(BABU_SEGMENT) Private	24	BABU_FIRST_ SEG	buffers for record data
(58)	ADDRESS Protected	4	BABU_NEXT_ SEG	address of next segment
(5C)	ADDRESS Protected	4	BABU_STG_ ADD	address of contents of segment
(60)	SIGNED Protected	4	BABU_STG_ LEN	length of storage in segment
(64)	SIGNED Protected	4	BABU_REC_ LEN	length of data in segment
(68)	SIGNED Protected	4	BABU_SEQ	segment number
(6C)	SIGNED Protected	4	BABU_FC_ UTOKEN	FC update token for segment
(70)	ADDRESS Private	4	BABU_WRITE_ STG_ADD	buffers for record data
(74)	ADDRESS Protected	4	PERMANENT_PTR	pointer to recoverable state
(78)	CHAR Protected	4	SOURCE_REF	
(78)	ADDRESS Protected	4	ACT_REQ_PTR	iff act_instore
(7C)	ADDRESS Protected	4	PARENT_ADD	Address of parent
(80)	ADDRESS Protected	4	NEXT	Chain pointers
(84)	ADDRESS Protected	4	PREV	used by EM browse

Constants

Len	Type	Value	Name	Description
4	DECIMAL	12	BACS_CONTAINER_ NOT_FOUND	
4	DECIMAL	11	BACS_LENGTH_ERROR	
4	DECIMAL	24	BACS_INVALID_ CONTAINER_NAME	
1	HEX	FF	HOP_TRUE	
1	HEX	00	HOP_FALSE	
4	DECIMAL	1	BALR_LENGTH_ERROR	
4	DECIMAL	2	BALR_IO_ERROR	
4	DECIMAL	3	BALR_DUPLICATE	
4	DECIMAL	4	BALR_BROWSE_END	
4	DECIMAL	5	BALR_FILE_ UNAVAILABLE	
4	DECIMAL	6	BALR_LOCKED	
4	DECIMAL	7	BALR_FILE_NOT_AUTH	
4	DECIMAL	8	BALR_RECORD_ NOT_FOUND	
4	DECIMAL	9	BALR_TIMEOUT	
4	DECIMAL	0	BALR_FIRST_ RECORD_NUMBER	
Uninit - just been allocated				
1	DECIMAL	0	BABU_STATE_UNINIT	
Init - filename, key and seg length known				
1	DECIMAL	1	BABU_STATE_INIT	

Len	Type	Value	Name	Description
Reading - after read_ record performed				
1	DECIMAL	2	BABU_STATE_READING	
Read - all bytes read (so it's been unflattened)				
1	DECIMAL	3	BABU_STATE_READ	
New - after Create_ Record				
1	DECIMAL	4	BABU_STATE_NEW	
Writing - after start_ write				
1	DECIMAL	6	BABU_STATE_WRITING	
Copied - after end_ write, mode=stor				
1	DECIMAL	5	BABU_STATE_COPIED	
Copied - after end_ write, mode=disk				
1	DECIMAL	7	BABU_STATE_WRITTEN	
1	DECIMAL	1	BABU_MODE_UNKN	
1	DECIMAL	2	BABU_MODE_DISK	
1	DECIMAL	3	BABU_MODE_COPY	
4	DECIMAL	1	BABU_WRITE_FAILURE	
4	DECIMAL	2	BABU_READ_FAILURE	
4	DECIMAL	3	BABU_FILE_ UNAVAILABLE	
4	DECIMAL	4	BABU_LOCKED	
4	DECIMAL	5	BABU_FILE_NOT_AUTH	
4	DECIMAL	6	BABU_KEY_NOT_FOUND	
4	DECIMAL	7	BABU_DUPLICATE	
4	DECIMAL	8	BABU_RECORD_BUSY	
4	DECIMAL	16384	BABU_MAX_SEG_LEN	
4	DECIMAL	60	BABU_HEADER_LEN	
4	DECIMAL	0	CMODE_INITIAL	not run/linked
4	DECIMAL	1	CMODE_RUN	run/linked
4	DECIMAL	2	CMODE_COMPLETE	completed

-

The length occupied by an Activity Set Element in a repository record is currently set as 80 bytes. This leaves some space should the data in the flat form of the object need to increase.

A dummy based variable is declared to provide a compile time check that the flat length is sufficient to accomodate the real object.

4	DECIMAL	80	FLAT_SET_ELEMENT_LENGTH	length occupied in records
1	DECIMAL	1	MODE_INITIAL	
1	DECIMAL	2	MODE_ACTIVE	
1	DECIMAL	3	MODE_DORMANT	
1	DECIMAL	4	MODE_CANCELLING	
1	DECIMAL	5	MODE_COMPLETE	
1	DECIMAL	1	COMPLETION_RESP_INCOMPLETE	
1	DECIMAL	2	COMPLETION_RESP_NORMAL	
1	DECIMAL	3	COMPLETION_RESP_FORCED	
1	DECIMAL	4	COMPLETION_RESP_ABEND_R	
1	DECIMAL	1	FIRE_REQUEST	
1	DECIMAL	2	DISPATCH_REQUEST	
abend_ request NOW UNUSED constant(3)				
1	DECIMAL	4	CANCEL_REQUEST	
1	DECIMAL	5	DELETE_REQUEST	
1	DECIMAL	0	RR_UNKNOWN	
1	DECIMAL	1	RR_FIRE_COMPL	
1	DECIMAL	2	RR_FIRE_INPUT	
1	DECIMAL	3	RR_FIRE_TIMER	
1	DECIMAL	5	RR_DELETE_CMD	
1	DECIMAL	6	RR_DELETE_COMPL	
1	DECIMAL	7	RR_DELETE_RESET	
1	DECIMAL	8	RR_DELETE_TREE	
1	DECIMAL	9	RR_CANCEL_CMD	
1	DECIMAL	10	RR_CANCEL_COMPL	
1	DECIMAL	11	RR_CANCEL_FORCE	
1	DECIMAL	12	RR_REATTACH_ACQ	
1	DECIMAL	1	EXEC_ASYNCHRONOUS	
1	DECIMAL	2	EXEC_SYNCHRONOUS	
2	CHARACTER	A	BAAC_ACTIVITY_RECORD_TYPE	

Len	Type	Value	Name	Description
-				

The length occupied by an Activity in a repository record is currently set as 400 bytes. This leaves some space should the data in the flat form of the object need to increase.

4	DECIMAL	400	FLAT_ACTIVITY_ LENGTH	

A dummy based variable is declared to provide a compile time check that the flat length is sufficient to accomodate the real object.

4	DECIMAL	64	FLAT_ACTIVITY_ SPARE	

BAACT Bam container class

-

What follows defines the Business Application Manager Container class.

-

Protect against multiple inclusion.

--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	48	CONTAINER	

--
-

An instance of the Container class consists of...

INSTANCE DATA				
Declared Data				
(0)	CHAR Protected	41	INSTANCE_ DATA_BLOCK	
(0)	OBJECT IsA(HOP_DCHAINNODE) Protected	16	CHAIN_LINK	chain linkage
Inherited Data				
(0)	CHAR Private	4	*	
(8)	CHAR Protected	8	*	
(8)	ADDRESS Protected	4	PREV	
(C)	ADDRESS Protected	4	NEXT	
(10)	CHAR Protected	16	CONTAINER_NAME	identifier
(20)	SIGNED Protected	4	DATA_LENGTH	amount of data
(24)	ADDRESS Protected	4	DATA_ADDRESS	address of data
(28)	BITSTRING Protected	1	CONTAINER_FLAGS	various flags
	`1... .... Protected`		FREE_HEADER	freemain flags
	`.111 1111 Protected`		*	- reserved

--
-

Class Data for the Container Class is declared as a private type. Storage for it is obtained for a single instance of the type from BADM during initialisation. BADM also looks after addressing it (via badm_set/inq_class_data).

SHARED DATA				
Declared Data				
(0)	CHAR Protected	48	BACO_CLASS_ DATA_TYPE	
(0)	STRUCTURE IsA(EYE_CATCHER_TYPE) Protected	16	EYE_CATCHER	eye catcher
(0)	UNSIGNED Protected	2	EYE_LEN	object length

Offset Hex	Type	Len	Name (Dim)	Description
(2)	UNSIGNED Protected	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHAR Protected	12	EYE_STRING	'>DFHddxxxxxx'
(10)	CHAR Protected	32	*	spare space for APARs
(0)	CHAR Protected	1024	BACO_SEGMENT_ TYPE	
(0)	CHAR Protected	8	BACO_SEGMENT_ HEADER	
(0)	ADDRESS Protected	4	BACO_NEXT_ SEGMENT	addr of next segment
(4)	SIGNED Protected	2	BACO_SEGMENT_ LEN	segment storage length
(6)	BITSTRING Protected	1	*	flags
	`1... ....` Protected		BACO_FREE_ SEGMENT	segment must be freed
	`.111 1111` Protected		*	reserved
(7)	CHAR Protected	1	*	reserved
(8)	CHAR Protected	1016	BACO_SEGMENT_ DATA	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	11	BACO_LENGTH_ERROR	
4	DECIMAL	1024	BACO_MAX_ SEGMENT_LEN	

BAAR Bam audit record class

Offset Hex	Type	Len	Name (Dim)	Description
4	DECIMAL	1	AF_DEF_PRO	
4	DECIMAL	2	AF_RUN_PRO	
4	DECIMAL	3	AF_LNK_PRO	
4	DECIMAL	4	AF_ACQ_PRO	
4	DECIMAL	5	AF_RST_PRO	
4	DECIMAL	6	AF_CAN_PRO	
4	DECIMAL	7	AF_SUS_PRO	
4	DECIMAL	8	AF_RES_PRO	
4	DECIMAL	9	AF_PUT_PRO	
4	DECIMAL	10	AF_DEL_PRO	
4	DECIMAL	11	AF_ACTIVATE	
4	DECIMAL	12	AF_COMPLETE	
4	DECIMAL	13	AF_LNK_ACT	
4	DECIMAL	14	AF_DEF_ACT	
4	DECIMAL	15	AF_RUN_ACT	
4	DECIMAL	16	AF_ACQ_ACT	
4	DECIMAL	17	AF_RST_ACT	
4	DECIMAL	18	AF_CAN_ACT	
4	DECIMAL	19	AF_SUS_ACT	
4	DECIMAL	20	AF_RES_ACT	
4	DECIMAL	21	AF_DEL_ACT	
4	DECIMAL	22	AF_DEF_TIM	
4	DECIMAL	23	AF_DEL_TIM	
4	DECIMAL	23	AF_MAX_FUNC	
4	DECIMAL	1	AR_RELEASE_1	
Reason Codes				
4	DECIMAL	62192	LOG_DISABLED	
4	DECIMAL	62193	LOG_NOT_FOUND	
4	DECIMAL	62194	LOG_IS_SYSTEM_LOG	
4	DECIMAL	62195	WRITE_ERROR	
4	DECIMAL	62196	LOG_STATUS_INVALID	
Message Numbers				
4	DECIMAL	101	MNO_XX01	
4	DECIMAL	102	MNO_XX02	

BAPT Bam processtype class

-

What follows defines the Business Application Manager Processtype class.

-

Protect against multiple inclusion.

--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	32	PROCESSTYPE	

--
-

An instance of the Container class consists of...

INSTANCE DATA				
Declared Data				
(0)	CHAR Protected	27	INSTANCE_ DATA_BLOCK	
(0)	CHAR Protected	8	NAME	identifier
(8)	CHAR Protected	8	FILE	file name
(10)	CHAR Protected	8	LOG	auditlog name
(18)	FIXED Protected	1	LEVEL	level of auditing
(19)	UNSIGNED Protected	1	USERRECS	user recs allowed
(1A)	FIXED Protected	1	STATUS	enabled or disabled
SHARED DATA				
Declared Data				
(0)	FIXED Public	1	ENABLESTATUS	
(0)	FIXED Public	1	AUDITLEVEL	

-

Class Data for the Processtype Class is declared as a private type. Storage for it is obtained for a single instance of the type from BADM during initialisation. BADM also looks after addressing it (via badm_ set/inq_ class_data).

(0)	CHAR Protected	52	BAPT_CLASS_ DATA_TYPE	
(0)	STRUCTURE IsA(EYE_CATCHER_TYPE) Protected	16	EYE_CATCHER	eye catcher
(0)	UNSIGNED Protected	2	EYE_LEN	object length
(2)	UNSIGNED Protected	2	EYE_OFFSET	offset of eye-catcher in object
(4)	CHAR Protected	12	EYE_STRING	'>DFHddxxxxxx'
(10)	CHAR Protected	4	PTT_DIRECTORY_ TOKEN	token for PTT
(14)	CHAR Protected	32	*	

Constants

Len	Type	Value	Name	Description
1	DECIMAL	0	ES_DISABLED	
1	DECIMAL	1	ES_ENABLED	
1	DECIMAL	0	AL_OFF	
1	DECIMAL	1	AL_PROCESS	
1	DECIMAL	2	AL_ACTIVITY	
1	DECIMAL	3	AL_FULL	
4	DECIMAL	17	NO_MORE_DATA	
4	DECIMAL	18	NOT_DISABLED	
4	DECIMAL	30	BA_CATALOG_ERROR	
4	DECIMAL	31	BA_DIRECTORY_ERROR	
4	CHARACTER	PTTE	PT_BLOCK_ NAME_VALUE	
8	CHARACTER	PTYPE	CATLG_TYPE	
14	CHARACTER	>DFHBAVPClass	EYE_CATCHER	

BRDCC Bridge control blocks

-

Lifetime of this storage is cics lifetime. It is created by DFHAPSI.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	64	BRSA	
(0)	UNSIGNED	4	BRSA_LENGTH	
(4)	CHARACTER	8	BRSA_EYE_CATCHER	>DFHBRSA
(C)	CHARACTER	4	*	reserved
(10)	UNSIGNED	4	BRSA_BFB_INDEX	Last value used in token
(14)	ADDRESS	4	BRSA_BFB_ KEEP_CHAIN	BFB keep chain anchor
(18)	CHARACTER	8	*	reserved
(20)	CHARACTER	8	BRSA_GENERAL_ SUBPOOL	General subpool
(28)	CHARACTER	8	BRSA_BRPC_ SUBPOOL	BRPC subpool token
(30)	CHARACTER	8	BRSA_BSB_SUBPOOL	BSB subpool token
(38)	CHARACTER	8	*	reserved
(40)	CHARACTER		*	

--
-

CICS key, task lifetime storage. It is created by DFHBRXM for init primary client.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	160	BRTA	
(0)	CHARACTER	16	BRTA_HEADER	
(0)	UNSIGNED	4	BRTA_LENGTH	
(4)	CHARACTER	8	BRTA_EYE_ CATCHER	>DFHBRTA
(C)	CHARACTER	4	*	reserved
(10)	CHARACTER	96	BRTA_BRIDGE_ ENVIRONMENT	
(10)	CHARACTER	4	BRTA_BRIDGE_ TRANSACTION_ID	Bridge transaction
(14)	CHARACTER	1	BRTA_CONTEXT	Bridge context
(15)	UNSIGNED	1	BRTA_CALL_ EXIT_FOR_SYNC	Call for syncpoint @P3C
(16)	CHARACTER	1	BRTA_FLAGS	
	1...		BRTA_LOAD_ ADS_DESCRIPTOR	Load ADSDs
	.1..		BRTA_BREXIT_ INIT_OK	

Offset Hex	Type	Len	Name (Dim)	Description
				Init call to brexit OK
	..11 1111		*	
(17)	CHARACTER	1	*	reserved
(18)	CHARACTER	2	BRTA_START_CODE	Start code
(1A)	CHARACTER	2	*	reserved
(1C)	CHARACTER	4	BRTA_TRANSACTION_ ID	User transaction id
(20)	CHARACTER	8	BRTA_USERID	Current userid
(28)	CHARACTER	8	BRTA_BREXIT_ PROGRAM	Bridge exit
(30)	CHARACTER	8	BRTA_FORMATTER_ PROGRAM	Bridge exit formatter
(38)	CHARACTER	8	*	reserved
(40)	CHARACTER	48	BRTA_IDENTIFIER	Value return on INQ TASK
(70)	CHARACTER	16	BRTA_FACILITY	
(70)	CHARACTER	8	BRTA_FACILITY_ TOKEN	Bridge Facility Token
(78)	ADDRESS	4	BRTA_BFB_PTR	-> Bridge Facility Block
(7C)	CHARACTER	4	BRTA_ORIGINAL_ NEXT_TRANID	Value in BFB on alloc
(80)	CHARACTER	32	BRTA_CONTROL_ BLOCKS	
(80)	ADDRESS	4	BRTA_BRDATA_PTR	-> BRDATA
(84)	FULLWORD	4	BRTA_BRDATA_LEN	Length BRDATA
(88)	ADDRESS	4	BRTA_BRXA_PTR	-> BRXA
(8C)	FULLWORD	4	BRTA_BRXA_LEN	Length BRXA
(90)	ADDRESS	4	BRTA_BRPC_PTR	-> BRPC
(94)	FULLWORD	4	BRTA_BRPC_LEN	Length BRPC
(98)	CHARACTER	8	*	reserved
(A0)	CHARACTER		*	

--
-

Lifetime of this storage is cics lifetime.

If the attach is done using DFHBRAT, the BRPC is create by DFHBRAT on an attach, and destroyed by DFHBRRM on transaction completion.

If it is created as part of piggy backing by another XM client the bridge will take a copy of the information. Ideally we should create the BRPC using a macro as this brakes domain rules. However the domains are tightly coupled so this is not an urgent problem.

We have a brpc_ version in case the primary client is ever shipped.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	64	BRPC	
(0)	CHARACTER	64	BRPC_PREFIX	
(0)	UNSIGNED	4	BRPC_LENGTH	Length of prefix+user data
(4)	CHARACTER	8	BRPC_EYE_ CATCHER	>DFHBRPC
(C)	UNSIGNED	4	BRPC_VERSION	0
(10)	CHARACTER	4	BRPC_BRIDGE_ TRANSACTION_ID	Bridge transaction
(14)	CHARACTER	1	BRPC_FLAGS	Bridge Flags @D1A
	1...		BRPC_TAKE_COPY	Piggy back copy @D1A
	.111 1111		*	reserved @D1A
(15)	CHARACTER	3	*	reserved
(18)	CHARACTER	8	BRPC_BREXIT_ PROGRAM	Bridge exit
(20)	CHARACTER	8	BRPC_USERID	Userid
(28)	CHARACTER	8	*	reserved
(30)	CHARACTER	12	*	reserved
(3C)	UNSIGNED	4	BRPC_BRDATA_LEN	length of user data
(40)	CHARACTER		BRPC_BRDATA	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	8192	BR_BFB_CATALOGUE_ INTERVAL	
1	DECIMAL	1	BRTA_CONTEXT_ NORMAL	not bridge environment
1	DECIMAL	2	BRTA_CONTEXT_ BRIDGE	bridge environment
1	DECIMAL	3	BRTA_CONTEXT_ BREXIT	running bridge exit
1	DECIMAL	1	BRTA_YES	
1	DECIMAL	2	BRTA_NO	

CAUTR CICS affinities utility trace table

-

What follows defines the CAUTrace class.

The CAUTrace class manages a trace table. This trace table is MVS GETMAINed. Each trace entry added to the table is a fixed length, 32 bytes. There are three formats of trace entry (see the signatures associated with method AUTR_PUT for further details). Each contains a time stamp. Since the trace table wraps once it been completely filled up, the time stamp can be used to determine the newest entry.

The CAUTrace class supplies the following basic methods:

- AUTR_CREATE - create and initialise the trace table.

- AUTR_DESTROY - destroy the trace table.

- AUTR_PUT - write a trace entry to the table.

- AUTR_GET_SIZE - return the size of the table.

See the method signatures for further details.

The CAUTrace class is used by the Transaction Affinities Utility to trace events and errors that occur while the utility is executing. Normal CICS tracing cannot be used by the CAU exit programs as it imposes too high an overhead and may cause loss of control. The trace table is output as part of a CICS system dump (use the parameter AU on the VERBEXIT to format the table) and as part of a transaction dump if the abend is associated with the Affinities Utility, that is, if the abend code is of the form AUxx.

-

The CAUTrace class has no instance data as there are no instances of this class. All data is stored in class data and is accessed by class methods. It only has internal methods.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	4	CAUTRACE	
INSTANCE DATA				
Declared Data				
(0)	CHAR Private	4	*	

-

The CAUTrace class data consists of the wrap-around trace table, the current position in the table, and an eye-catcher.

Offset Hex	Type	Len	Name (Dim)	Description
SHARED DATA				
Declared Data				
(0)	CHAR Protected	32032	CLASSDATABLOCK	
(0)	STRUCTURE IsA(EYECATCHER) Protected	16	EYE_CATCHER	an eye-catcher
(0)	UNSIGNED Public	2	EYE_LEN	object length
(2)	UNSIGNED Public	2	EYE_OFFSET	offset of eye-catcher
(4)	CHAR Public	12	EYE_STRING	eye-catcher string

Offset Hex	Type	Len	Name (Dim)	Description
(4)	CHAR Public	1	EYE_LT	>
(5)	CHAR Public	3	EYE_PFX	DFH
(8)	CHAR Public	8	EYE_NAME	AUTR
(10)	ADDRESS Protected	4	CURRENT_ POSITION	position in table
(14)	CHAR Protected	12	*	reserved
(20)	CHAR Protected	32	TRACE_TABLE (1000)	trace table
(20)	STRUCTURE IsA(CAFFEVENTENTRY) Protected	32	CAFF_EVENT_ ENTRY	caff event trace
(20)	CHAR Protected	5	CAFF_EVENT_ MODULE	last 5 chars of modname
(25)	CHAR Protected	1	CAFF_EVENT_ SPACE	blank space
(26)	CHAR Protected	18	CAFF_EVENT_ TEXT	text
(38)	CHAR Protected	8	CAFF_EVENT_ TIME	timestamp
(20)	STRUCTURE IsA(EXITEVENTENTRY) Protected	32	EXIT_EVENT_ ENTRY	exit event trace
(20)	CHAR Protected	5	EXIT_EVENT_ MODULE	last 5 chars of modname
(25)	CHAR Protected	1	EXIT_EVENT_ SPACE	blank space
(26)	CHAR Protected	14	EXIT_EVENT_ TEXT	text
(34)	CHAR Protected	4	EXIT_EVENT_ TASKNUM	task number
(38)	CHAR Protected	8	EXIT_EVENT_ TIME	timestamp
(20)	STRUCTURE IsA(EXITERRORENTRY) Protected	32	EXIT_ERROR_ ENTRY	exit error trace
(20)	CHAR Protected	5	EXIT_ERROR_ MODULE	last 5 chars of modname
(25)	CHAR Protected	1	EXIT_ERROR_ SPACE	blank space
(26)	CHAR Protected	6	EXIT_ERROR_ TEXT	text
(2C)	CHAR Protected	4	EXIT_ERROR_ TASKNUM	task number
(30)	UNSIGNED Protected	4	EXIT_ERROR_ TM_TABLE	cautabm table number
(34)	UNSIGNED Protected	1	EXIT_ERROR_ TM_FUNCTION	cautabm function
(35)	UNSIGNED Protected	1	EXIT_ERROR_ TM_RESPONSE	cautabm response
(36)	UNSIGNED Protected	1	EXIT_ERROR_ TM_REASON	cautabm reason
(37)	UNSIGNED Protected	1	*	
(38)	CHAR Protected	8	EXIT_ERROR_ TIME	timestamp

--
-

Declare associated types. There are types for eye catcher, the different types of trace entry (which must be 32 characters long in total), store clock, responses.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Public	16	EYECATCHER	eye-catcher type
(0)	UNSIGNED Public	2	EYE_LEN	object length
(2)	UNSIGNED Public	2	EYE_OFFSET	offset of eye-catcher
(4)	CHAR Public	12	EYE_STRING	eye-catcher string
(4)	CHAR Public	1	EYE_LT	>
(5)	CHAR Public	3	EYE_PFX	DFH
(8)	CHAR Public	8	EYE_NAME	AUTR
(0)	CHAR Protected	8	STORECLOCK	
(0)	CHAR Public	18	CAFFEVENTTEXT	
(0)	CHAR Public	14	EXITEVENTTEXT	
(0)	CHAR Public	6	EXITERRORTEXT	
(0)	CHAR Protected	32	CAFFEVENTENTRY	caff event trace entry
(0)	CHAR Protected	5	CAFF_EVENT_ MODULE	last 5 chars of modname
(5)	CHAR Protected	1	CAFF_EVENT_ SPACE	blank space
(6)	STRUCTURE IsA(CAFFEVENTTEXT) Protected	18	CAFF_EVENT_TEXT	text
(18)	STRUCTURE IsA(STORECLOCK) Protected	8	CAFF_EVENT_TIME	timestamp
(0)	CHAR Protected	32	EXITEVENTENTRY	exit event trace entry
(0)	CHAR Protected	5	EXIT_EVENT_ MODULE	last 5 chars of modname
(5)	CHAR Protected	1	EXIT_EVENT_ SPACE	blank space

Offset Hex	Type	Len	Name (Dim)	Description
(6)	STRUCTURE IsA(EXITEVENTTEXT) Protected	14	EXIT_EVENT_TEXT	text
(14)	CHAR Protected	4	EXIT_EVENT_ TASKNUM	task number
(18)	STRUCTURE IsA(STORECLOCK) Protected	8	EXIT_EVENT_TIME	timestamp
(0)	CHAR Protected	32	EXITERRORENTRY	exit error trace entry
(0)	CHAR Protected	5	EXIT_ERROR_ MODULE	last 5 chars of modname
(5)	CHAR Protected	1	EXIT_ERROR_ SPACE	blank space
(6)	STRUCTURE IsA(EXITERRORTEXT) Protected	6	EXIT_ERROR_TEXT	text
(C)	CHAR Protected	4	EXIT_ERROR_ TASKNUM	task number
(10)	UNSIGNED Protected	4	EXIT_ERROR_ TM_TABLE	cautabm table number
(14)	UNSIGNED Protected	1	EXIT_ERROR_ TM_FUNCTION	cautabm function
(15)	UNSIGNED Protected	1	EXIT_ERROR_ TM_RESPONSE	cautabm response
(16)	UNSIGNED Protected	1	EXIT_ERROR_ TM_REASON	cautabm reason
(17)	UNSIGNED Protected	1	*	
(18)	STRUCTURE IsA(STORECLOCK) Protected	8	EXIT_ERROR_TIME	timestamp
(0)	FIXED Public	1	TRRESPONSE	

--
-

Declare registers used by this class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	SIGNED Protected	4	R0	
(0)	SIGNED Protected	4	R1	
(0)	SIGNED Protected	4	R2	
(0)	SIGNED Protected	4	RE	
(0)	SIGNED Protected	4	RF	

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	AUTR_OK	
1	DECIMAL	2	AUTR_EXCEPTION	
1	DECIMAL	3	AUTR_DISASTER	
1	DECIMAL	6	AUTR_PURGED	

CCGD Catalog static storage

Module Name = DFHCCGD
DESCRIPTIVE NAME = CICS/MVS Catalog Global Definitions.
Function =
These are the common definitions for DFHCCCC and DFHCCDM
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = n/a
Module Type = n/a
Attributes = n/a
Storage
Catalog's storage consists of :
"Static" storage, which is GETMAINed during DFHCCDM
initialisation and lasts until FREEMAINed during
DFHCCDM termination.
This storage is DECLAREd in this copybook, which is
included in DFHCCCC and DFHCCDM. This storage contains
the anchor block.
Automatic storage which is acquired each time a call is
made to DFHCCCC or DFHCCDM.
This storage is defined by the DECLAREs made in DFHCCCC
and DFHCCDM.
Catalog's anchor block
based on anchor CCANCHORP, double word aligned.
anchor defined in DFHKERN TYPE(DOMENTER)
storage GETMAINed during catalog's initialization
Catalog's static storage based on CCANCHORP, double word
aligned.
1. Area whose size is known at PL/AS compile time.
Pointers to ACB, array of RPLs, array of buffers.
Catalog's status variables
Array of per thread variables
2. Areas whose size is not known until assemble time
Array of buffers (one per thread)
ACB
Array of RPLs (one per thread)
Macro parameter settings
MAX_DATA_LENGTH must be set to the length used when the
DFHCCD dataset was defined, minus the length of the VSAM key.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	2652	CCANCHORB	CC's static stg
(0)	HALFWORD	2	CC_STATIC_LEN	Length of cc's static storage
(2)	CHARACTER	14	CC_ANC_ EYECATCHER	eyecatcher
(2)	CHARACTER	1	CC_ANC_ARROW	'>'
(3)	CHARACTER	3	CC_ANC_DFH	'DFH'
(6)	CHARACTER	2	CC_ANC_DOMID	'LC' or 'GC'
(8)	CHARACTER	8	CC_ANC_ BLOCK_NAME	'ANCHOR '
(10)	CHARACTER	8	*	type of catalog
(10)	FULLWORD	4	CATALOG_TYPE	DFHCC_DOMAIN | DFHGC_DOMAIN
(14)	CHARACTER	2	TYPE_CATALOG	"LC" or "GC"
(16)	UNSIGNED	1	CAT_TYPE_ME	1=local , 2=global for ME
(17)	CHARACTER	1	*	
Catalog's global status				
(18)	ADDRESS	4	BUFFER_ARRAY_A	start of array of Buffers
(1C)	ADDRESS	4	VSAM_ACB_A	a(VSAM_ACB)
(20)	ADDRESS	4	RPL_ARRAY_A	start of array of RPLs
(24)	ADDRESS	4	OPEN_PLIST_A	Open parameter list
(24)	BITSTRING	1	*	
	1...		CCSOPLMO	end marker for plist-os
(28)	ADDRESS	4	CC_SER_ LOCK_TOKEN	lock_token
(2C)	HALFWORD	2	ENVIRONMENT	CC to use CICS | OS macros
(2E)	BITSTRING	1	CC_STRING_ WAIT_ECB	USED IN WAIT_OLDC CALL
(2F)	UNSIGNED	1	OPEN_STATUS	File is OPEN | CLOSED
(30)	CHARACTER	1	RESERVED	Reserved
(34)	FULLWORD	4	NUM_THREADS	Number of VSAM strings
(38)	FULLWORD	4	MAX_DATA_LENGTH	max data size for catalog
(3C)	CHARACTER	8	CC_SER_LOCK	Serialization lock name
(44)	BITSTRING	1	*	
	1...		CATALOG_ACTIVE	Catalog is initialized and not yet terminated.
	.111 1111		*	Reserved
(45)	CHARACTER	3	*	Reserved
(48)	FULLWORD	4	CC_STARTUP_TOKEN	Token used in startup
(4C)	ADDRESS	4	CC_STARTUP_TASK	task id of startup task
Per thread storage				
(50)	CHARACTER	80	STRING_STORAGE (32)	Per thread array

Offset Hex	Type	Len	Name (Dim)	Description
(50)	CHARACTER	8	STRING_ EYECATCHER	"CCTHREAD" | "GCTHREAD"
	RPL and Buffer addresses.			
(58)	ADDRESS	4	STRING_RPL_A	RPL address
(5C)	ADDRESS	4	STRING_BUFFER_A	Address of buffer in STRING_STORAGE array
(60)	ADDRESS	4	STRING_VSAM_RECORD_A	Address of record in VSAM buffer (Provided by vsam)
	State of this thread			
(64)	FULLWORD	4	STRING_TOKEN	NB 0 = thread is free
(68)	ADDRESS	4	STRING_ XC_WAIT_ECB	Wait ECB for vsam exclusive control
(6C)	CHARACTER	1	STRING_STATES	THREAD STATUS
	1...		STRING_XC	Holding Exclusive control
	.1..		WAIT_XC	Waiting on Exclusive control
	..1.		ENDREQ_XC	Endreq required during xc
	...1 1111		*	reserved
(6D)	UNSIGNED	1	STRING_FUNCTION	Function request at connect
	Browsing parameters			
(6E)	HALFWORD	2	STRING_ BROWSE_RC	RC from START_BROWSE
(70)	CHARACTER	28	STRING_KEY	Full KEY
(70)	CHARACTER	12	STRING_ DOM_TYPE	start-browse DOM.TYPE
(70)	CHARACTER	4	STRING_DOM	calling DOM
(74)	CHARACTER	8	STRING_TYPE	TYPE
(7C)	CHARACTER	16	STRING_NAME	NAME
	Keep request to vsam and RPL feedback for debug			
(8C)	CHARACTER	4	STRING_ VSAM_DEBUG	To debug vasm problems
(8C)	CHARACTER	1	STRING_VSAM_REQUEST	last RPL request byte
(8D)	CHARACTER	3	STRING_RPL_FEEDBACK	last RPL feedback info
	Dump diagnostic information for problem analysis			
(90)	CHARACTER	4	STRING_TRANSID	Thread owner tranid
(94)	CHARACTER	4	STRING_TASKNUM	Thread owner taskno
(98)	CHARACTER	8	*	Reserved
(A50)	FULLWORD	4	SEQ_WRITE_NUMBER	Sequential write attempts@P4A
(A54)	FULLWORD	4	NOSEQ_WRITE_ NUMBER	Non-seq write attempts
(A58)	FULLWORD	4	SEQ_RETRY_NUMBER	Number seq writes failed

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1	VPLOPT1	OPTION byte 1 in VSAM RPL
	1...		VPLLOC	1=Locate mode. 0=Move mode
	.1..		VPLDIR	1=Direct access
	..1.		VPLSEQ	1=Sequential access
	...1		VPLSKP	1=Skip sequential access
	.... 1...		VPLASY	1=Asynchronous processing 0=Synchronous processing
	.... .1..		VPLKGE	1=Search KEY >= 0=Search KEY equal
	.... ..1.		VPLGEN	1=Generic KEY request 0=Full KEY search argument
	1		VPLECBSW	1=External ECB

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1	VPLOPT2	OPTION byte 2 in VSAM RPL
	1...		VPLKEY	1=Locate record by KEY
	.1..		VPLADR	1=Addressed access = RPLADD
	..1.		VPLCNV	1=Control interval access
	...1		VPLBWD	1=Bwd. 0=Fwd
	.... 1...		VPLLRD	1=LRD last record ... 0=ARD User's argument...
	.... .1..		VPLWAITX	1=aynch proc wait 0=never take exit
	.... ..1.		VPLUPD	1=Update request
	1		VPLNSP	1=Note string position

String buffers defined, one per thread

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	STRING_BUFFER	Will be based on STRING_BUFFER_A(token)
(0)	CHARACTER	28	STRING_ BUFFER_KEY	VSAM key
(0)	CHARACTER	12	STRING_BUFFER_DOM_TYPE	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER	4	STRING_ BUFFER_DOM	DOM.TYPE for browse domain
(4)	CHARACTER	8	STRING_ BUFFER_TYPE	type
(C)	CHARACTER	16	STRING_ BUFFER_NAME	name
(1C)	CHARACTER	*	STRING_ BUFFER_DATA	file data

Constants

Len	Type	Value	Name	Description
2	HEX	2B10	TRID_CC_ADD_LEN	Data too long
2	HEX	2B70	TRID_CC_DATA_ TOO_LONG	Read cmds
2	HEX	2010	TRID_CC_ENTRY	CCCC
2	HEX	2050	TRID_CC_EXIT	CCCC
2	HEX	2020	TRID_CC_EXTENT	New vsam extent
2	HEX	2B20	TRID_CC_FUNCTION	CCCC
2	HEX	2B71	TRID_CC_PUT_R_LEN	Too long
2	HEX	2B30	TRID_CC_RECOVERY	CCCC
2	HEX	2070	TRID_CC_SERIAL_ ENTRY	CCCC
2	HEX	2080	TRID_CC_SERIAL_ EXIT	CCCC
2	HEX	2B40	TRID_CC_ST_ WAIT_UNLOCK	CCCC
2	HEX	2B41	TRID_CC_ST_ WAIT_LOCK	CCCC
2	HEX	2B42	TRID_CC_CHANGE_ MODE	CCCC
2	HEX	2B43	TRID_CC_RESTORE_ MODE	CCCC
2	HEX	2B44	TRID_CC_WAIT_OLDC	CCCC
2	HEX	2B50	TRID_CC_TOKEN	CCCC bad token
2	HEX	2B52	TRID_CC_TOKEN2	END-BROWSE bad T
2	HEX	2B53	TRID_CC_TOKEN3	END-WRITE bad T
2	HEX	2B54	TRID_CC_TOKEN4	GET-NEXT bad token
2	HEX	2B55	TRID_CC_TOKEN5	PUT-REPLACE bad T
2	HEX	2B56	TRID_CC_TOKEN6	WRITE-NEXT bad T
2	HEX	2B57	TRID_CC_TOKEN7	DELETE bad T
2	HEX	2B58	TRID_CC_TOKEN8	STARTUP_O dup
2	HEX	2B59	TRID_CC_TOKEN9	no STARTUP_OP
2	HEX	2B5A	TRID_CC_NOT_ FOR_LCD	only GCD
2	HEX	2B5B	TRID_CC_USE_ WRITE_N	use write_next for startup
2	HEX	2B5C	TRID_CC_USE_TOKEN	alloc tok
2	HEX	2B60	TRID_CC_VSAM	CCCC
2	HEX	20A0	TRID_CC_VSAM_END	CCCC
2	HEX	2090	TRID_CC_VSAM_WAIT	CCCC
2	HEX	2B73	TRID_CC_WR_NX_LEN	too long
2	HEX	2B72	TRID_CC_WRITE_LEN	too long
2	HEX	20C0	TRID_CC_XC_ WAIT_LOCK	CCCC
2	HEX	20B0	TRID_CC_XC_ WAIT_UNLOCK	CCCC
2	HEX	1B50	TRID_DM_ADD_LOCK	CCDM
2	HEX	1010	TRID_DM_ENTRY	CCDM
2	HEX	1040	TRID_DM_EXIT	CCDM
2	HEX	1020	TRID_DM_RECOVERY	CCDM
2	HEX	1B40	TRID_DM_SET_PHASE	CCDM
2	HEX	1B60	TRID_DM_UNLOCK	CCDM
2	HEX	1B30	TRID_DM_VSAM_ERROR	CCDM
Constants				
8	CHARACTER	CCSERLCK	CC_LOCK	Serialization (local)
2	DECIMAL	2	CICS	CICS environment
2	CHARACTER	CC	COMPONENT_ID	"CC" is "component"
8	CHARACTER	GCSERLCK	GC_LOCK	Serialization (local)
2	DECIMAL	1	XA	XA environment
0	BIT	1	COND	COND=YES
0	BIT	0	FALSE	boolean
1	DECIMAL	0	FILE_CLOSED	CC FILE is CLOSED
1	DECIMAL	1	FILE_OPEN	CC FILE is OPEN
2	DECIMAL	28	KEY_LENGTH	Size of vsam KEY bin caller id. size in bytes user's TYPE field size user's NAME field size
2	CHARACTER	GC	GLOBAL_CATALOG	Type of catalog
1	DECIMAL	2	GLOBAL_ME	Global catalog ME insert
2	CHARACTER	CC	LOCAL_CATALOG	Type of catalog
1	DECIMAL	1	LOCAL_ME	Local catalog ME insert
1	DECIMAL	0	OK	good return code value
4	DECIMAL	0	THREAD_FREE	string is free
0	BIT	1	TRUE	boolean
0	BIT	0	UNCOND	COND=NO
0	BIT	0	WAIT	Wait bit value for ECB
0	BIT	1	WAIT_END	End-wait bit value for ECB
VSAM request codes				
1	HEX	00	VSAMGET	VSAM get
1	HEX	01	VSAMPUT	VSAM put
1	HEX	02	VSAMCHEK	VSAM check

Len	Type	Value	Name	Description
1	HEX	03	VSAMPNT	VSAM point
1	HEX	04	VSAMEREQ	VSAM endreq
1	HEX	05	VSAMERAS	VSAM erase

CPCPS Cpi-c conversation control block

```
CONTROL BLOCK NAME = DFHCPCPS
DESCRIPTIVE NAME = CICS/ESA
               CPI-C Conversation Control Block (CPC)
               & log data records
FUNCTION =
    To provide CPI-C's principal control block record structure
    There is one instance of a CPC per CPI-C conversation.
    A CPC contains conversation identifier and control
    information relating to its CPI-C conversation.
    At various stages during the lifetime of a CPI-C
    conversation the CPC will be associated with a session
    control block (TCTTE) which will act as the CPI-C
    conversations principal facility for communication.
LIFETIME =
    The lifetime of a single CPI-C conversation
STORAGE CLASS =
    The CPC will exist in CICS main (31bit) storage.
LOCATION =
    All CPCs associated with a single task are chained from
    the system TCA at TCACPCCN.
INNER CONTROL BLOCKS =
    A further record definition is included in this copybook
    for CPIC_LOG_DATA. This control block is addressed via
    a pointer in the CPC named "log_data_buffer_ptr".
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS = None
 MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
     None
  DATA AREAS =
     None
  CONTROL BLOCKS =
     TCTTE (via an associated session control block pointer)
  GLOBAL VARIABLES (Macro pass) =
     None
        R E A D T H I S N O T I C E F I R S T
This PL/AS object has been commented using the ABSTRACT tool.
Please make sure any changes you make are consistent with the
use of this tool. Either use ABSTRACT to view the file, or avoid
deleting any of the open/close comment folds.
( The following record defines the structure of the
| CPI-C Conversation Control Block (CPC)
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	212	DFHCPCPS	
	(... control block header and chaining information			
(0)	UNSIGNED	2	CPC_RECORD_ LENGTH	
(2)	CHARACTER	14	CPC_EYECATCHER	
	identifier for this conversation			
(10)	CHARACTER	8	CONVERSATION_ID	
	pointer to next CPC in chain for this task			
(18)	ADDRESS	4	NEXT_CPC_PTR	
	session tctte for this cpi-c conversation			
(1C)	ADDRESS	4	TCTTE_PTR	
	) (... conversation characteristics these are parameters that may or must be set before certain cpi-c calls may be made for this conversation			
(20)	UNSIGNED	4	CONVERSATION_ TYPE	
(24)	UNSIGNED	4	DEALLOCATE_TYPE	
(28)	UNSIGNED	4	ERROR_DIRECTION	
(2C)	UNSIGNED	4	LOG_DATA_LENGTH	
(30)	ADDRESS	4	LOG_DATA_ BUFFER_PTR	
(34)	UNSIGNED	4	FILL	
(38)	UNSIGNED	4	MODE_NAME_LENGTH	
(3C)	CHARACTER	8	MODE_NAME	

Offset Hex	Type	Len	Name (Dim)	Description
(44)	UNSIGNED	4	PARTNER_LU_NAME_LENGTH	
(48)	CHARACTER	17	PARTNER_LU_NAME	
(59)	CHARACTER	7	*	
(60)	UNSIGNED	4	PREPARE_TO_RECEIVE_TYPE	
(64)	UNSIGNED	4	RECEIVE_TYPE	
(68)	UNSIGNED	4	RETURN_CONTROL	
(6C)	UNSIGNED	4	SEND_TYPE	
(70)	UNSIGNED	4	SYNC_LEVEL	
(74)	UNSIGNED	4	TP_NAME_LENGTH	
(78)	CHARACTER	64	TP_NAME	

) (... other conversation related information these CPC fields are required by this CPI-C implementation to store certain items of information across calls to the interface

Offset Hex	Type	Len	Name (Dim)	Description
(B8)	UNSIGNED	4	CONVERSATION_ STATE	
(BC)	CHARACTER	8	PROFILE_NAME	
(C4)	BITSTRING	1	*	
	1...		NEXT_LL_CONCATENATED	
	.1..		ID_NOT_RECEIVED	
	..1.		PARTIAL_ ID_RECEIVED	
	...1 1111		*	
(C5)	CHARACTER	1	PARTIAL_ID	
(C6)	BITSTRING	1	*	
	1...		NEW_STATE_AFTER_BACKOUT_ RULES	
	.111 1111		*	
(C7)	BITSTRING	1	*	
(C8)	UNSIGNED	4	OUTSTANDING_ LL_COUNT	
(CC)	UNSIGNED	4	STATE_AFTER_ COMMIT	
(D0)	UNSIGNED	4	SYNCPOINT_RETURN_CODE	

)
(The following record defines the structure used
| to contain conversation related log data for CPI-C
It is addressed via a pointer in the CPC.
It is followed by a constant defining the offset
of the log data itself in the structure.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	CPIC_LOG_DATA	
(0)	UNSIGNED	2	LOG_DATA_RECORD_LENGTH	
(2)	CHARACTER	14	LOG_DATA_ EYECATCHER	
(10)	UNSIGNED	4	LOG_DATA_BUFFER_LENGTH	
(14)	CHARACTER	*	LOG_DATA	

Constants

Len	Type	Value	Name	Description
2	DECIMAL	20	LOG_DATA_HDR_LEN	

CPSPS Cpi static storage area

```
CONTROL BLOCK NAME = DFHCPSPS
DESCRIPTIVE NAME = CICS CPI Static Storage Area
FUNCTION =
    This control block provides the global information for
    the CPI which must be around for the duration of the CICS
    execution.
    It contains:
      CPI initialization suspend token
      CPI status
      Entry points of CPI modules
      CPI-C last conversation-id
LIFETIME =
    The control block is created during CICS initialisation
    by DFHSIB1, and exists for as long as the CICS system.
STORAGE CLASS =
    The control block is in subpool DFHAPDANY
LOCATION =
    The CPI Static Area is located by field SSZCPI in
    DFHSSAPS
INNER CONTROL BLOCKS = None
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS = None
 MODULE TYPE = Control block definition
EXTERNAL REFERENCES = None
  DATA AREAS = None
  CONTROL BLOCKS = None
  GLOBAL VARIABLES (Macro pass) = None
    CPI STATIC STORAGE AREA
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	44	CPI_SSA	
Block prefix				
(0)	CHARACTER	16	PREFIX	block prefix area
(0)	HALFWORD	2	BLOCK_LENGTH	block length
(2)	CHARACTER	1	ARROW	'>'
(3)	CHARACTER	3	DFH	'DFH'
(6)	CHARACTER	2	DOMID	'CP'
(8)	CHARACTER	8	BLOCK_NAME	'CPSTATIC'
Block body				
(10)	CHARACTER	28	BODY	body of block
CPI fields				
(10)	CHARACTER	8	*	
(10)	ADDRESS	4	INIT_SUSPEND_ TOKEN	Suspend token
(14)	UNSIGNED	1	INIT_STATUS	CPI Initialization status
(15)	CHARACTER	3	*	Reserved
CPI module entry points				
(18)	CHARACTER	12	*	
(18)	ADDRESS	4	DFHCPARH_ADDR	DFHCPARH entry point
(1C)	ADDRESS	4	DFHCPSRH_ADDR	DFHCPSRH entry point
(20)	ADDRESS	4	DFHCPIR_ADDR	DFHCPIR entry point
CPI-C static storage				
(24)	CHARACTER	8	*	
(24)	CHARACTER	8	CPIC_LAST_ CONVID	Last conversation-id used by CPI-C

Constants

Len	Type	Value	Name	Description
1	DECIMAL	44	CPI_SSA_LENGTH	
Constants representing status of CPI initialisation				
2	DECIMAL	1	CPI_STATIC_ STORAGE_INITIALIZED	
2	DECIMAL	2	CPI_ACQUIRE_ SUSPEND_TOK_FAILED	
2	DECIMAL	3	CPI_ACQUIRED_ SUSPEND_TOK	
2	DECIMAL	4	CPI_INIT_ TASK_ATTACHED	
2	DECIMAL	5	CPI_INIT_ TASK_STARTED	
2	DECIMAL	6	CPI_LOAD_ CPIC_FAILED	
2	DECIMAL	7	CPI_LOADED_CPIC	
2	DECIMAL	8	CPI_LOAD_ CPIRR_FAILED	
2	DECIMAL	9	CPI_LOADED_CPIRR	
2	DECIMAL	10	CPI_INIT_SUCCEEDED	
2	DECIMAL	11	CPI_OPEN_ FOR_BUSINESS	
Block name for CP static				
8	CHARACTER	CPSTATIC	CPI_SSA_BLOCK_ NAMEI	

DDBSC Directory manager building blocks

AVL2 Header structure for instance:
AVLTREE

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	AVL2	
(0)	CHARACTER	12	DUMMY	Unused
(C)	ADDRESS	4	ROOT	Pointer to root
(10)	ADDRESS	4	FRST	Pointer to first
(14)	ADDRESS	4	LAST	Pointer to last
(18)	FULLWORD	4	NOEL	Number of elements
(1C)	FULLWORD	4	ELEN	Element length

End of AVL2 Header structure

AVL2 Node structure for instance:
AVLTREE

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	NODE	
(0)	CHARACTER	16	HDR	
(0)	ADDRESS	4	LEFT	Left child
(4)	ADDRESS	4	RITE	Right child
(8)	ADDRESS	4	PAPA	Parent
(C)	FULLWORD	4	BFAC	Balancing factor
(10)	CHARACTER	*	DATA	Data portion

DDCBC Directory manager structures

Directory Manager Domain Structures and Constants.
The Directory manager anchor block and other internal directory structures are described below.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	DDA	

The Directory Manager Anchor Block

(0)	CHARACTER	16	DDA_PREFIX	
(0)	HALFWORD	2	DDA_LENGTH	Structure length
(2)	CHARACTER	1	DDA_ARROW	>
(3)	CHARACTER	3	DDA_DFH	DFH
(6)	CHARACTER	2	DDA_DOMID	DD
(8)	CHARACTER	8	DDA_BLOCK_NAME	ANCHOR
(10)	CHARACTER	8	DDA_IDIRECTORYCLASS	
(10)	ADDRESS	4	DDA_DIRECTORY_ LIST	Directory header chain
(14)	UNSIGNED	1	DDA_STATE	Directory Manager state
(15)	CHARACTER	3	*	Reserved
(18)	CHARACTER	32	DDA_CICS_BITS	
(18)	CHARACTER	8	DDA_GENERAL_ SUBPOOL	Directory general subpool
(20)	CHARACTER	8	DDA_BROWSE_ SUBPOOL	Directory browse subpool
(28)	ADDRESS	4	DDA_GLOBAL_LOCK	Directory global lock
(2C)	BITSTRING	1	*	
	1...		DDA_COLD_START	Was it a cold start
(2D)	CHARACTER	3	*	Reserved
(30)	CHARACTER	4	*	Reserved
(34)	CHARACTER	4	*	Reserved
(38)	CHARACTER		DDA_END	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	72	DIRHEAD	

A Directory Header structure.
There is one of these for each directory. It is created by the Directory manager CREATE_ DIRECTORY function, and is chained on to the list of directories in alphabetical order. It lasts until CICS terminates.

(0)	CHARACTER	16	DH_PREFIX	
(0)	HALFWORD	2	DH_LENGTH	Structure length
(2)	CHARACTER	1	DH_ARROW	>
(3)	CHARACTER	3	DH_DFH	DFH
(6)	CHARACTER	2	DH_DOMID	DD
(8)	CHARACTER	8	DH_BLOCK_NAME	DIR_HEAD
(10)	CHARACTER	20	DH_CICS_BITS	
(10)	ADDRESS	4	DH_NEXT	Next directory in chain
(14)	ADDRESS	4	DH_PREV	Previous directory in chain
(18)	ADDRESS	4	DH_LOCAL_LOCK	Directory local lock
(1C)	CHARACTER	8	DH_SUBPOOL	Fixed length subpool
(24)	CHARACTER	8	DH_IDIRECTORY	
(24)	CHARACTER	4	DH_DIRNAME	Directory name
(28)	FULLWORD	4	DH_DIRKEYLENGTH	Key length (4 to 252)

The Lookup Map section of the Directory Header.
This holds the information for fast location of an entry name

(2C)	CHARACTER	16	DH_ILOOKUPMAP	
(2C)	FULLWORD	4	DH_HASHSIZE	Size of the hash table
(30)	FULLWORD	4	DH_HASHELEMS	Current number of entries
(34)	ADDRESS	4	DH_HASHTABLE	Address of hash table
(38)	BITSTRING	1	*	
	1...		DH_REHASH	Rehash required flag
(39)	CHARACTER	3	*	Reserved

The Browse Seq section of the Directory Header.
This holds the information used for browsing the directory

(3C)	CHARACTER	12	DH_IBROWSESEQ	
(3C)	FULLWORD	4	DH_DELETES	Number of deletes
(40)	ADDRESS	4	DH_CURRENT_ BROWSES	Browses on this directory
(44)	ADDRESS	4	DH_BROWSETREE	The browse tree
(48)	CHARACTER		DH_END	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	HASHELEM	

A hash chain element.
One exists for each entry name in each directory. It is created by the ADD_ ENTRY function, and is chained on to the collision list from the hash table. It is destroyed by the DELETE_ ENTRY function.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	ADDRESS	4	HE_NEXT	Next on collision list
(4)	CHARACTER	8	HE_TOKEN	Corresponding data token
(C)	CHARACTER		HE_NAME	Variable length key name

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	HASHSTRUCT	

The hash table structure.
There is one of these for each directory, created either by the CREATE_ DIRECTORY function, or by the ADD_ ENTRY function when performing a dynamic re-hash. It is destroyed during a dynamic re-hash.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER	16	HS_PREFIX	
(0)	HALFWORD	2	HS_LENGTH	Structure length
(2)	CHARACTER	1	HS_ARROW	>
(3)	CHARACTER	3	HS_DFH	DFH
(6)	CHARACTER	2	HS_DOMID	DD
(8)	CHARACTER	8	HS_BLOCK_NAME	HASH_TBL
(10)	CHARACTER		HS_HASHTABLE	The actual hash table

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	292	BROWSE_VAL	

This structure holds the information for a browse on a particular directory.
The structure is created by the Directory manager START_ BROWSE function, and is chained on to the list of current browses if not in task_ related storage.
It is destroyed by the END_ BROWSE function, or if task_ related, at end-of-task.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER	16	BV_PREFIX	
(0)	HALFWORD	2	BV_LENGTH	Structure length
(2)	CHARACTER	1	BV_ARROW	>
(3)	CHARACTER	3	BV_DFH	DFH
(6)	CHARACTER	2	BV_DOMID	DD
(8)	CHARACTER	8	BV_BLOCK_NAME	BRWS_VAL
(10)	ADDRESS	4	BV_NEXT	Next browse_val in list
(14)	ADDRESS	4	BV_PREV	Previous browse_val
(18)	FULLWORD	4	BV_OLDDELETES	Deletes after get next
(1C)	ADDRESS	4	BV_OLDCURSOR	Cursor after get next
(20)	BITSTRING	1	BV_FLAGS	
	1...		BV_ON_NAME	Are we on a name yet
	.1..		BV_DONE_GETNEXT	Have we done a getnext
	..1.		BV_TASK_RELATED	Task-related browse?
(21)	CHARACTER	3	*	Reserved
(24)	CHARACTER	256	BV_OLDNAME	Name after last get next
(124)	CHARACTER		BV_END	

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	PREINITIALISING	
1	DECIMAL	2	PREINITIALISED	
1	DECIMAL	3	INITIALISED	
1	DECIMAL	4	QUIESCED	
1	DECIMAL	5	TERMINATED	

The valid range of values for the key length.

Len	Type	Value	Name	Description
4	DECIMAL	4	MINKEYLEN	Minimum key length
4	DECIMAL	252	MAXKEYLEN	Maximum key length

General constants used by Directory Manager.

Len	Type	Value	Name	Description
8	CHARACTER	DDGENRAL	DD_GENERAL_SP	
8	CHARACTER	DDBROWSE	DD_BROWSEVAL_SP	
8	CHARACTER	DDGLOCK	DD_GLOBAL_LOCK	
4	CHARACTER	DDL_	DD_LOCK_PREFIX	
4	CHARACTER	DDS_	DD_SUBPOOL_PREFIX	
1	CHARACTER	>	ARROW	
3	CHARACTER	DFH	DFH	
8	CHARACTER	ANCHOR	BLOCKNAME_DDA	

Len	Type	Value	Name	Description
8	CHARACTER	HASH_TBL	BLOCKNAME_HS	
8	CHARACTER	HASHELEM	BLOCKNAME_HE	
8	CHARACTER	DIR_HEAD	BLOCKNAME_DH	
8	CHARACTER	BRWS_VAL	BLOCKNAME_BV	
8	CHARACTER	AVL_NODE	BLOCKNAME_AN	
8	CHARACTER	AVL_HEDR	BLOCKNAME_AH	
2	CHARACTER	DD	COMPID	
8	CHARACTER	DD HSIZE	DD_CATALOG_TYPE	
0	BIT	1	TRUE	
0	BIT	0	FALSE	

DEGPC Dce services domain global statistics

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	DFHDEGPS	de global stats
(0)	HALFWORD	2	DEG_STATS_LENGTH	length of record
(2)	HALFWORD	2	DEG_STATS_ID	de global stats id, should contain deg_dcl_id
(4)	UNSIGNED	1	DEG_STATS_ VERSION	de global stats version
(5)	UNSIGNED	3	*	filler
(8)	FULLWORD	4	DEG_THREADS	number of DCE threads available for use
(C)	FULLWORD	4	DEG_QUEUED_REQS	number of requests currently queued
(10)	FULLWORD	4	DEG_QUEUE_ HIWATER	peak number of requests queued
(14)	FULLWORD	4	DEG_PROCESSING_ REQS	number of requests being processed
(18)	FULLWORD	4	DEG_REQS_HIWATER	peak number of requests being processed
(1C)	FULLWORD	4	DEG_TOTAL_ REQS_RCVD	total requests received by DCE
(20)	FULLWORD	4	DEG_TOTAL_ REQS_DEQ	total requests allocated to a thread
(24)	FULLWORD	4	DEG_TOTAL_ REQS_PROCESS	total requests processed by DCE (i.e. completed)
(28)	CHARACTER	8	DEG_TOTAL_ REQS_P_TIME	total time requests spent being processed
(30)	CHARACTER	8	DEG_TOTAL_ REQS_Q_TIME	total time requests spent on the queue

Constants

Len	Type	Value	Name	Description
1	HEX	01	DEG_DCL_VERSION	version number
2	DECIMAL	83	DEG_DCL_ID	DE global id statistics id

DHANC Document handler anchor block

-

This anchor block contains the global storage for the DH domain.

It defines the domain state information, variables and constants required by the DH gates and other external programs such as DFHDHTRI, the domain trace interpretation routine.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	128	DHA	

-

Block header

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER	16	DHA_PREFIX	===> eyecatcher <===
(0)	HALFWORD	2	DHA_LENGTH	length of dha
(2)	CHARACTER	14	DHA_PREFIX_TEXT	>DFHDHAnchor

--
-

Domain state information

Offset Hex	Type	Len	Name (Dim)	Description
(10)	UNSIGNED	1	DHA_DH_STATE	DH domain state initialised, quiesced or terminated
(11)	UNSIGNED	1	DHA_FLAGS	
	1...		DHA_COLD_START	CICS cold started
	.1..		DHA_XRSINDI_ ACTIVE	XRSINDI exit active
(12)	CHARACTER	1	*	Reserved
(13)	UNSIGNED	1	DHA_DEFAULT_CODEPAGE_LEN	Length of codepage
(14)	FULLWORD	4	DHA_NUM_ DOCUMENTS	Number of documents
(18)	CHARACTER	8	DHA_DEFAULT_CODEPAGE	Default codepage
(20)	ADDRESS	4	DHA_LOCK_TOKEN	DH domain lock token
(24)	ADDRESS	4	DHA_TLD_ LOCK_TOKEN	Template lock token
(28)	STRUCTURE IsA(ETOKEN)	8	DHA_GENERAL_ SPTOKEN	General subpool token
(28)	ADDRESS	4	P	
(2C)	FULLWORD	4	N	
(30)	STRUCTURE IsA(ETOKEN)	8	DHA_DBB_SPTOKEN	DBB subpool token
(30)	ADDRESS	4	P	
(34)	FULLWORD	4	N	
(38)	STRUCTURE IsA(ETOKEN)	8	DHA_DCB_SPTOKEN	DCB subpool token
(38)	ADDRESS	4	P	
(3C)	FULLWORD	4	N	
(40)	STRUCTURE IsA(ETOKEN)	8	DHA_DCR_SPTOKEN	DCR subpool token
(40)	ADDRESS	4	P	
(44)	FULLWORD	4	N	
(48)	STRUCTURE IsA(ETOKEN)	8	DHA_DDB_SPTOKEN	DDB subpool token
(48)	ADDRESS	4	P	
(4C)	FULLWORD	4	N	
(50)	STRUCTURE IsA(ETOKEN)	8	DHA_DOA_SPTOKEN	DOA subpool token
(50)	ADDRESS	4	P	
(54)	FULLWORD	4	N	
(58)	STRUCTURE IsA(ETOKEN)	8	DHA_STB_SPTOKEN	STB subpool token
(58)	ADDRESS	4	P	
(5C)	FULLWORD	4	N	
(60)	STRUCTURE IsA(ETOKEN)	8	DHA_TLD_SPTOKEN	TLD subpool token
(60)	ADDRESS	4	P	
(64)	FULLWORD	4	N	
(68)	ADDRESS	4	DHA_TLD_DHT1_DIRTOKEN	DHT1 directory token
(6C)	ADDRESS	4	DHA_TLD_DHT2_DIRTOKEN	

Offset Hex	Type	Len	Name (Dim)	Description
				DHT2 directory token
(70)	CHARACTER	8	DHA_TEMPLATE_ DCB_CHAIN	
				DCB descriptor chain
(70)	ADDRESS	4	DHA_PDS_ DCB_FIRST	First DCB descriptor
(74)	ADDRESS	4	DHA_PDS_ DCB_LAST	Last DCB descriptor

--

Offset Hex	Type	Len	Name (Dim)	Description
(78)	ADDRESS	4	DHA_FIRST_DOA	
(7C)	ADDRESS	4	DHA_LAST_DOA	
(80)	CHARACTER		DHA_END	

-

DH Domain Document Anchor Block

Document anchor block - 1 per transaction

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	40	DOA	
(0)	CHARACTER	16	DOA_PREFIX	
(0)	HALFWORD	2	DOA_LENGTH	
(2)	CHARACTER	1	DOA_ARROW	>
(3)	CHARACTER	3	DOA_DFH	DFH
(6)	CHARACTER	2	DOA_DOMID	DH
(8)	CHARACTER	8	DOA_BLOCK_NAME	DOA
(10)	ADDRESS	4	DOA_NEXT	-> next document anchor
(14)	ADDRESS	4	DOA_PREV	-> previous document anchor
(18)	ADDRESS	4	DOA_FIRST_DCR	-> first document ctl rec
(1C)	ADDRESS	4	DOA_LAST_DCR	-> last document ctl rec
(20)	CHARACTER	4	DOA_TRANNUM	Transaction number
(24)	CHARACTER	4	DOA_TRANSID	Transaction id
(28)	CHARACTER		*	

--
-

DH Domain Document Control Record

Document control record - 1 per document

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	92	DCR	
(0)	CHARACTER	16	DCR_PREFIX	
(0)	HALFWORD	2	DCR_LENGTH	
(2)	CHARACTER	1	DCR_ARROW	>
(3)	CHARACTER	3	DCR_DFH	DFH
(6)	CHARACTER	2	DCR_DOMID	DH
(8)	CHARACTER	8	DCR_BLOCK_NAME	DCR
(10)	ADDRESS	4	DCR_NEXT	-> next document ctl rec
(14)	ADDRESS	4	DCR_PREV	-> previous document ctl rec
(18)	ADDRESS	4	DCR_FIRST_CELEM	-> first doc content element
(1C)	ADDRESS	4	DCR_LAST_CELEM	-> last doc content element
(20)	ADDRESS	4	DCR_FIRST_DBP	-> first document bookmark
(24)	ADDRESS	4	DCR_LAST_DBP	-> last document bookmark
(28)	FULLWORD	4	DCR_DOCUMENT_ COUNT	counter used in document token
(2C)	FULLWORD	4	DCR_DOCUMENT_ SIZE	total size of export document
(30)	FULLWORD	4	DCR_NUM_BKMARKS	number of document bookmarks
(34)	FULLWORD	4	DCR_NUM_DATABLKS	number of document data blocks
(38)	FULLWORD	4	DCR_NUM_SYMBOLS	number of symbols
(3C)	FULLWORD	4	DCR_DATA_SIZE	size of document data
(40)	FULLWORD	4	DCR_SYMBOL_SIZE	size of symbol data
(44)	CHARACTER	12	DCR_SYMBOL_ MANAGER	Building block access vars
(44)	ADDRESS	4	DCR_SYMBOL_ TABLE	Hash table locator
(48)	ADDRESS	4	DCR_SYMBOL_ STORAGE_MGR	
				Symbol storage locator
(4C)	ADDRESS	4	DCR_SYMBOL_ BLOCK_MGR	
				Symbol block manager
(50)	ADDRESS	4	DCR_FIRST_ TEMPLATE	-> first template on chain
(54)	ADDRESS	4	DCR_LAST_ TEMPLATE	-> last template on chain
(58)	FULLWORD	4	DCR_EMBED_DEPTH	Template embed depth
(5C)	CHARACTER		*	

--
-

DH Domain Document Data Block

Document data block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	DDB	
(0)	CHARACTER	16	DDB_PREFIX	
(0)	HALFWORD	2	DDB_LENGTH	
(2)	CHARACTER	1	DDB_ARROW	>
(3)	CHARACTER	3	DDB_DFH	DFH
(6)	CHARACTER	2	DDB_DOMID	DH
(8)	CHARACTER	8	DDB_BLOCK_NAME	DDB
(10)	ADDRESS	4	DDB_NEXT_CELEM	-> next doc content element
(14)	ADDRESS	4	DDB_PREV_CELEM	-> prev doc content element
(18)	BITSTRING	1	*	
	1...		DDB_NONBIN_ BLOCK	Content is non-binary data
	.1..		DDB_BIN_BLOCK	Content is binary data
	..11 1111		*	
(19)	CHARACTER	3	*	For alignment
(1C)	CHARACTER	8	DDB_CODEPAGE	Data host codepage
(24)	FULLWORD	4	DDB_DATA_LENGTH	Length of data portion
(28)	CHARACTER	*	DDB_DATA	Data block value

--
-

DH Domain Document Bookmark Block

Document bookmark block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	DBB	
(0)	CHARACTER	16	DBB_PREFIX	
(0)	HALFWORD	2	DBB_LENGTH	
(2)	CHARACTER	1	DBB_ARROW	>
(3)	CHARACTER	3	DBB_DFH	DFH
(6)	CHARACTER	2	DBB_DOMID	DH
(8)	CHARACTER	8	DBB_BLOCK_NAME	DBB
(10)	ADDRESS	4	DBB_NEXT_CELEM	-> next doc content element
(14)	ADDRESS	4	DBB_PREV_CELEM	-> prev doc content element
(18)	BITSTRING	1	*	
	11..		*	
	..1.		DBB_BOOKMARK	Content is bookmark
	...1 1111		*	
(19)	CHARACTER	3	*	For alignment
(1C)	ADDRESS	4	DBB_NEXT_BKMARK	-> next document bookmark
(20)	ADDRESS	4	DBB_PREV_BKMARK	-> previous document bookmark
(24)	CHARACTER	16	DBB_BKMARK_NAME	Bookmark name

--
-

DH Domain Document Template Block

Document template block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	DTB	
(0)	ADDRESS	4	DTB_NEXT_ TEMPLATE	-> next doc template block
(4)	ADDRESS	4	DTB_PREV_ TEMPLATE	-> prev doc template block
(8)	CHARACTER		DTB_TEMPLATE_ DATA	Template data

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	DH_STATE_ INITIALISING	
1	DECIMAL	2	DH_STATE_ INITIALISED	
1	DECIMAL	3	DH_STATE_QUIESCING	
1	DECIMAL	4	DH_STATE_QUIESCED	
1	DECIMAL	5	DH_STATE_ TERMINATED	

--
-

Literals

Len	Type	Value	Name	Description
8	CHARACTER	DHGENRAL	SPNAME_GENERAL	General purpose subpool for DH domain
8	CHARACTER	DHDOA	DH_DOA_SP	Document anchor block subpool
8	CHARACTER	DHDCR	DH_DCR_SP	Document control record subpool
8	CHARACTER	DHDBB	DH_DBB_SP	Document bookmark block subpool
8	CHARACTER	DHSTB	DH_STB_SP	Symbol table block subpool
8	CHARACTER	DHDDB	DH_DDB_SP	Document data subpool
14	CHARACTER	>DFHDHANCHOR	DHA_EYE_CATCHER	
8	CHARACTER	DHLOCK	DH_LOCK_NAME	Domain lock

--
-

Error codes (for DFHKERN RECOVERY_REQUEST)

Len	Type	Value	Name	Description
4	CHARACTER	ADHA	LOCK_ERROR_CODE	
4	CHARACTER	ADHB	UNLOCK_ERROR_CODE	

--
-

Constants used for symbol table subpool initialisation

Len	Type	Value	Name	Description
4	DECIMAL	8192	DH_STB_LENGTH	

DHTL Document handler template descriptor

-

Document Domain Template Descriptor.

This control block is the internal representation of one instance of a Document Handler domain template definition, or DOCTEMPLATE.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	128	DFHDHTLC	
(0)	CHARACTER	16	DHTL_PREFIX	Standard eyecatcher
(0)	HALFWORD	2	DHTL_LENGTH	
(2)	CHARACTER	1	DHTL_ARROW	
(3)	CHARACTER	3	DHTL_DFH	
(6)	CHARACTER	2	DHTL_DOMID	
(8)	CHARACTER	8	DHTL_BLOCK_NAME	
(10)	CHARACTER	8	DHTL_DOCTEMPLATE	Name of RDO DOCTEMPLATE
(18)	CHARACTER	48	DHTL_TEMPLATE_ NAME	Full name of template
(48)	CHARACTER	2	DHTL_TEMPLATE_ TYPE	Type of template
(4A)	BITSTRING	1	DHTL_TEMPLATE_ FLAGS	Properties flags
	1...		DHTL_APPEND_ CRLF	Append crlf to recs
	.1..		DHTL_TYPE_ BINARY	Template is bin
	..1.		DHTL_TYPE_ EBCDIC	Template is ebcdic
	...1 1111		*	Reserved
(4B)	UNSIGNED	1	*	Reserved
(4C)	BITSTRING	4	*	Reserved
(50)	CHARACTER	48	DHTL_TEMPLATE_ BODY	Type-specific overlay
(50)	CHARACTER	8	DHTL_RESOURCE_ NAME	Generic resource name
(50)	CHARACTER	48	DHTL_PDS_ DESCRIPTOR	PDS-member type template
(50)	CHARACTER	44	DHTL_BLDL_DATA	Data returned by BLDL
(50)	CHARACTER	8	DHTL_MEMBER_ NAME	Member name
(58)	UNSIGNED	3	DHTL_MEMBER_ TTR	TTR of member
(5B)	UNSIGNED	1	DHTL_CONCATENATION_ NO	Concatenation set by BLDL
(5C)	UNSIGNED	1	DHTL_LIBRARY_ TYPE	Library type set by BLDL
(5D)	UNSIGNED	1	DHTL_MEMBER_ LEN	Length of directory data
(5E)	CHARACTER	30	DHTL_MEMBER_ DATA	ISPF-editor-specific data
(5E)	UNSIGNED	1	DHTL_MEMBER_ VERSION	Version number of member
(5F)	UNSIGNED	1	DHTL_MEMBER_ MODLEVEL	Modification level
(60)	UNSIGNED	2	*	Reserved
(62)	BITSTRING	4	DHTL_MEMBER_ DATE1	Creation date of member
(66)	BITSTRING	4	DHTL_MEMBER_ DATE2	Last update date
(6A)	BITSTRING	2	DHTL_MEMBER_ HHMM	Last update time
(6C)	HALFWORD	2	DHTL_MEMBER_ CURRENT_SIZE	Curr lines in member
(6E)	HALFWORD	2	DHTL_MEMBER_ INITIAL_SIZE	Init lines in member
(70)	HALFWORD	2	DHTL_MEMBER_ MODLN	Number of modified lines
(72)	CHARACTER	8	DHTL_MEMBER_ USERID	Last update userid
(72)	CHARACTER	8	DHTL_DDNAME	Overlaid with ddname
(7A)	CHARACTER	2	*	Reserved
(7C)	ADDRESS	4	DHTL_PDS_ DCB_DESCRIPTOR	Pointer to DCB descriptor
(50)	CHARACTER	8	DHTL_FILE_ DESCRIPTOR	

Offset Hex	Type	Len	Name (Dim)	Description
				FILE type template
(50)	CHARACTER	8	DHTL_TEMPLATE_FILENAME	
				CICS filename
(50)	CHARACTER	8	DHTL_PROGRAM_DESCRIPTOR	
				PROGRAM type template
(50)	CHARACTER	8	DHTL_TEMPLATE_PGMNAME	
				CICS program name
(50)	CHARACTER	16	DHTL_TSQUEUE_DESCRIPTOR	
				TSQUEUE type template
(50)	CHARACTER	16	DHTL_TEMPLATE_TSQNAME	
				CICS TSQueue name
(50)	CHARACTER	4	DHTL_TDQUEUE_DESCRIPTOR	
				TDQUEUE type template
(50)	CHARACTER	4	DHTL_TEMPLATE_TDQNAME	
				CICS TDQueue name
(50)	CHARACTER	8	DHTL_EXITPGM_DESCRIPTOR	
				EXITPGM type template
(50)	CHARACTER	8	DHTL_TEMPLATE_EXITPGM	
				CICS EXITPGM name
(80)	CHARACTER		DHTL_TEMPLATE_ END	Alignment

-

This data area described the DCB structure that is used for reading partitioned datasets containing templates. Because it is the interface to the BPAM access method, it must reside below 16M.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	472	DFHDHPDC	
(0)	CHARACTER	16	DHPD_PREFIX	Standard eyecatcher
(0)	HALFWORD	2	DHPD_LENGTH	
(2)	CHARACTER	1	DHPD_ARROW	
(3)	CHARACTER	3	DHPD_DFH	
(6)	CHARACTER	2	DHPD_DOMID	
(8)	CHARACTER	8	DHPD_BLOCK_NAME	
(10)	ADDRESS	4	DHPD_DCB_NEXT	Pointer to next DCB entry
(14)	ADDRESS	4	DHPD_DCB_PREV	Pointer to prev DCB entry
(18)	CHARACTER	8	DHPD_DDNAME	DDNAME for template PDS
(20)	CHARACTER	8	DHPD_STATUS	
(20)	FULLWORD	4	*	Reserved
(24)	UNSIGNED	1	DHPD_FILETYPE	Type: 0=PDS 1=PDSE
(25)	BITSTRING	1	DHPD_FLAG1	Reserved for flags
(26)	HALFWORD	2	*	Reserved
(28)	CHARACTER	8	DHPD_DCB_ OPENLIST	Openlist
(28)	BITSTRING	1	*	Member DCB OPEN option
(29)	ADDRESS	3	DHPD_MEMBER_DCB_PTR	
				Member DCB address
(2C)	BITSTRING	1	*	Directory DCB OPEN option
(2D)	ADDRESS	3	DHPD_DIRECTORY_DCB_PTR	
				Directory DCB address
(30)	ADDRESS	4	DHPD_SYNAD_PTR	Ptr to 31-bit SYNAD
(34)	ADDRESS	4	DHPD_MEMBER_EODAD_PTR	
				Ptr to 31-bit EODAD
(38)	ADDRESS	4	DHPD_DIRECTORY_EODAD_PTR	
				Ptr to 31-bit EODAD
(3C)	ADDRESS	4	DHPD_ABEND_ EXIT_PTR	
				Ptr to 31-bit ABEXIT
(40)	CHARACTER	8	DHPD_EXIT_LIST	
(40)	UNSIGNED	1	DHPD_EXLST_ABEND_EXIT_CODE	
				Code for abend exit
(41)	ADDRESS	3	DHPD_EXLST_ABEND_EXIT_PTR	
				Ptr to abend exit
(44)	UNSIGNED	1	DHPD_EXLST_JFCB_CODE	

Offset Hex	Type	Len	Name (Dim)	Description
				Code for JFCB entry
(45)	ADDRESS	3	DHPD_EXLST_ JFCB_PTR	
				Ptr to JFCB
(48)	CHARACTER	24	DHPD_AMODE24_EXIT_ROUTINES	
(48)	BITSTRING	6	DHPD_IO_ ERROR_RTN	24-bit SYNAD stub routine
(4E)	BITSTRING	6	DHPD_MEMBER_EOD_RTN	
				24-bit EODAD stub routine
(54)	BITSTRING	6	DHPD_DIRECTORY_EOD_RTN	
				24-bit EODAD stub routine
(5A)	BITSTRING	6	DHPD_ABEND_ EXIT_RTN	
				24-bit ABEXIT stub
(60)	CHARACTER	24	DHPD_DECB	BPAM DECB
(78)	CHARACTER	88	DHPD_MEMBER_DCB	Member DCB
(D0)	CHARACTER	88	DHPD_DIRECTORY_ DCB	Directory DCB
(128)	CHARACTER	176	DHPD_JFCB	JFCB
(128)	CHARACTER	44	DHPD_DSNAME	Dataset name
(1D8)	CHARACTER		DHPD_DCB_DESCRIPTOR_END	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	0	DHPD_FILETYPE_PDS	Normal PDS
4	DECIMAL	1	DHPD_FILETYPE_PDSE	Extended PDS
4	DECIMAL	2	DHPD_FILETYPE_HFS	HFS file

DMAFC Dm authorised facility state

-

DFHDMAFC

DFHDMAFC is the copy book that defines the domain manager authorized facility state and interface.

The domain manager authorized facilities are provided to the CICS address space. This state is anchored in the AFCB.

When an ENFREQ ACTION=LISTEN request is issued MVS returns a token that uniquely identifies the listen request. This token must be specified on the ACTION=DELETE request. These tokens will be stored in key 0 storage to ensure that CICS will not delete some other subsystems listen requests. A slot in the AFCB will be required to anchor the domain manager key 0 state. The address of the AFCB will be passed as the PARM on the ENFREQ ACTION=LISTEN.

-

DMAF_STATE

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	DMAF_STATE	
(0)	UNSIGNED	2	DMAFS_LEN	
(2)	CHARACTER	14	DMAFS_EYE	
(10)	ADDRESS	4	DMAFS_ENF_ANCHOR	
(14)	ADDRESS	4	DMAFS_TCB	
(18)	ADDRESS	4	DMAFS_ASCB	
(1C)	BITSTRING	4	DMAFS_ENF_DTOKEN (1)	

--
-

DMAF_PLIST

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	20	DMAF_PLIST	
(0)	HALFWORD	2	DMAF_PLISTLEN	
(2)	BITSTRING	2	*	
(4)	UNSIGNED	1	DMAF_FUNCTION	
(5)	BITSTRING	1	*	
(6)	UNSIGNED	1	DMAF_RESPONSE	
(7)	UNSIGNED	1	DMAF_REASON	
(8)	BITSTRING	4	DMAF_ENF_REASON	
(C)	ADDRESS	4	DMAF_ENF_ANCHOR	
(10)	BITSTRING	4	DMAF_SVC_ RESPONSE	

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	DMAF_LISTEN	
1	DECIMAL	2	DMAF_DELETE	
1	DECIMAL	1	DMAF_OK	
1	DECIMAL	2	DMAF_EXCEPTION	
1	DECIMAL	3	DMAF_INVALID	
1	DECIMAL	4	DMAF_DISASTER	
1	DECIMAL	1	DMAF_GETMAIN_ D_FAIL	
1	DECIMAL	2	DMAF_GETMAIN_ S_FAIL	
1	DECIMAL	3	DMAF_FESTAE_FAIL	
1	DECIMAL	4	DMAF_NOT_AUTHED	
1	DECIMAL	5	DMAF_INVALID_ FUNCTION	
1	DECIMAL	6	DMAF_DUPLICATE_ REQUEST	
1	DECIMAL	7	DMAF_LISTEN_ INACTIVE	
1	DECIMAL	8	DMAF_LISTEN_ ENF_ERROR	
1	DECIMAL	9	DMAF_DELETE_ ENF_ERROR	
1	DECIMAL	10	DMAF_SVC_ CALL_A_FAIL	
1	DECIMAL	11	DMAF_SVC_ CALL_D_FAIL	

DMCB1 Domain manager anchor block

Segment Name = DFHDMCB1
DESCRIPTIVE NAME = CICS/MVS Domain Manager (DM)
Control Blocks 1.
Function =
This file contains the data structure
declarations used by the Domains Manager.
The data structure is :
ANCHOR - DM Anchor block
Also declared are the macro replacement
variables used by DM.
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
DM anchor block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1964	ANCHOR	Anchor Block
(0)	CHARACTER	16	ANC_PREFIX	Anchor prefix area
(0)	HALFWORD	2	ANC_LENGTH	Anchor length
(2)	CHARACTER	1	ANC_ARROW	Arrow eyecatcher
(3)	CHARACTER	3	ANC_DFH	DFH
(6)	CHARACTER	2	ANC_DOMID	Domain id
(8)	CHARACTER	8	ANC_BLOCK_NAME	Control block name
(10)	CHARACTER	1896	PHASE_MANAGEMENT	Phase Management
(10)	CHARACTER	16	PM_PREFIX	Phase manage. prefix area
(10)	HALFWORD	2	PM_LENGTH	Phase manage. length
(12)	CHARACTER	1	PM_ARROW	Arrow eyecatcher
(13)	CHARACTER	3	PM_DFH	DFH
(16)	CHARACTER	2	PM_DOMID	Domain id
(18)	CHARACTER	8	PM_BLOCK_NAME	Control block name
(20)	CHARACTER	2	*	Filler
(22)	HALFWORD	2	PM_PHASE_STATE	Global phase state
(24)	HALFWORD	2	PM_NO_ACTIVE_ DOMAINS	Number of active domains
(26)	HALFWORD	2	*	Filler
(28)	CHARACTER	52	PM_DOM_TABLE (36)	Array of domain information
(28)	FULLWORD	4	PM_DOMAIN_ TOKEN	Domain index
(2C)	CHARACTER	2	PM_DOMAIN_ID	Domain identifier
(2E)	HALFWORD	2	PM_ACT_PHASE	Actual phase of domain
(30)	BITSTRING	1	*	
	1...		PM_ACTIVE	'1' active, '0' inact
	.111 1111		*	Reserved
(31)	BITSTRING	3	*	Filler
(34)	CHARACTER	8	PM_TOTAL_ TIME_IN_QUEUE	Total time in q
(3C)	CHARACTER	8	PM_TIME_ STARTED_TO_INIT	Time started init
(44)	CHARACTER	8	PM_TIME_ INITIALISED	Time finished init
(4C)	CHARACTER	8	PM_TIME_ STARTED_ TO_QUIESCE	Time started quie
(54)	CHARACTER	8	PM_TIME_ QUIESCED	Time finished quie
(778)	CHARACTER	4	SYSTEM_ STATUS_COMMAND	System Status Command
(778)	BITSTRING	1	*	
	1...		SSC_INIT	'1' initialised/ing
	.1..		SSC_QUIESCE	'1' quiesced/ing
	..1.		SSC_TERM	'1' terminated/ing
	...1 1111		*	Reserved
(779)	BITSTRING	3	*	Filler
(77C)	CHARACTER	24	WQ_HEAD	Dummy wait queue element
(794)	CHARACTER	8	SUBPTOK	Subpool token
(794)	ADDRESS	4	SUBPTOK_P	-> to subpool token
(798)	FULLWORD	4	SUBPTOK_N	Length of token
(79C)	ADDRESS	4	LOCKTOK	Lock token
(7A0)	CHARACTER	3	INIT_STATS_COLL	Yes/No
(7A3)	CHARACTER	3	QUIESCE_ STATS_COLL	Yes/No
(7A6)	CHARACTER	2	*	reserved
(7A8)	ADDRESS	4	ENF_ANCHOR_ ADDRESS	A(ENF_ANCHOR)

Constants

Len	Type	Value	Name	Description
2	DECIMAL	2560	DMPH_TOP	
Language Environment is initialised				
2	DECIMAL	2484	DMPH_LANGUAGE_ ENVIRONMENT_READY	
Recovery_ active - Recovery Manager can now unshunt shunted units of work				
2	DECIMAL	2480	DMPH_RECOVERY_ ACTIVE	
System_ log_available - The CICS system log is now available for use				
2	DECIMAL	2475	DMPH_SYSTEM_ LOG_AVAILABLE	*
TS_basic_recovery_complete - Interval control can now make inquiries to TS about IC queues.				
2	DECIMAL	2473	DMPH_TS_BASIC_ RECOVERY_COMPLETE	
RM_ clients_ registered - Client registration completed				
2	DECIMAL	2470	DMPH_RM_CLIENTS_ REGISTERED	
Basic_ functions_ available - Basic functions can now be used				
2	DECIMAL	2432	DMPH_BASIC_ FUNCTIONS_AVAILABLE	
Statistics_ available - ap is ready for statistics to be collected during initialisation				
2	DECIMAL	2048	DMPH_STATISTICS_ AVAILABLE	
Global_ catalog_ available - the global catalog is ready for use				
2	DECIMAL	1536	DMPH_GLOBAL_ CATALOG_AVAILABLE	
RM_ startup_ type_known - RM has discovered the type of start				
2	DECIMAL	1312	DMPH_RM_STARTUP_ TYPE_KNOWN	
Global_ catalog_ for_RM - Catalog is available for RM only				
2	DECIMAL	1296	DMPH_GLOBAL_ CATALOG_FOR_RM	
Primary_ terminated - in the case of the Alternate, this means that the decision to take over has been finalised by XRF and its I/O has been prevented. In the case of the Primary this phase is 'skipped over'.				
2	DECIMAL	1280	DMPH_PRIMARY_ TERMINATED	
Default_ user_available - the default user has been added				
2	DECIMAL	1200	DMPH_DEFAULT_ USER_AVAILABLE	
ESM_ available - the ESM Signon function is available				
2	DECIMAL	1184	DMPH_ESM_AVAILABLE	
CWA_ available - the CWA is available				
2	DECIMAL	1168	DMPH_CWA_AVAILABLE	
XM_ attach_ available - Transaction Manager XMAT Attach available				
2	DECIMAL	1156	DMPH_XM_ATTACH_ AVAILABLE	
System_ functions_ available - all the services required by XM ATTACH are now available				
2	DECIMAL	1152	DMPH_SYSTEM_ FUNCTIONS_AVAILABLE	
CSA_ available - the CSA is available				
2	DECIMAL	1024	DMPH_CSA_AVAILABLE	
Timer_ available - the timer is ready for use				
2	DECIMAL	768	DMPH_TIMER_ AVAILABLE	
Pre_ init_complete - pre initialisation is complete, initialisation can proceed				
2	DECIMAL	512	DMPH_PRE_ INIT_COMPLETE	

Len	Type	Value	Name	Description
Quiesce Phases Shutdown_ stats_ready - the statistics domain will wait on this phase being set before taking shutdown statistics.				
2	DECIMAL	2304	DMPH_SHUTDOWN_ STATS_READY	
Statistics_ unavailable - the statistics domain has completed its last statistics collection and from now on no more statistics will be taken.				
2	DECIMAL	2048	DMPH_STATISTICS_ UNAVAILABLE	
Applications_ finished - all user transactions have finished				
2	DECIMAL	1792	DMPH_APPLICATIONS_ FINISHED	
Bottom - the system/domain has quiesced.				
2	DECIMAL	256	DMPH_BOTTOM	

DMCB2 Domain manager browse cursor

Segment Name = DFHDMCB2
DESCRIPTIVE NAME = CICS/MVS Domain Manager (DM)
Control Blocks 2.
Function =
This file contains data structure
declarations used by the Lock Manager domain.
The file is included by the inquiry module of the
Domain Manager (DM).
The data structure is :
BROWSE_CURSORS - DM Browsing details.
Also declared, are the macro replacement variables
used by DFHDMIQ.
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
Browse Cursors

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	20	BROWSE_CURSORS	Browse Cursors
(0)	CHARACTER	16	BC_PREFIX	BC Prefix area
(0)	HALFWORD	2	BC_LENGTH	BC length
(2)	CHARACTER	1	BC_ARROW	Arrow eyecatcher
(3)	CHARACTER	3	BC_DFH	DFH
(6)	CHARACTER	2	BC_DOMID	Domain id
(8)	CHARACTER	8	BC_BLOCK_NAME	Control block name
(10)	FULLWORD	4	BC_CURSOR	Cursor value
(14)	CHARACTER		*	Filler

DMCB3 Domain manager wait queue element

Segment Name = DFHDMCB3
DESCRIPTIVE NAME = CICS/MVS Domain Manager (DM)
Control Blocks 3.
Function =
This file contains data structure
declarations used by the Domain Manager.
The file is included by all Domain Manager modules.
The data structure is :
WAIT_QUEUE - DM Wait queue information
Subpool and lock token information is included by
DFHDMWQ only.
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
Wait queue

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(0)	STRUCTURE	40	WAIT_QUEUE	Wait Queue
(0)	CHARACTER	24	WQ_PREFIX	Wait queue prefix area
(0)	HALFWORD	2	WQ_LENGTH	Length
(2)	CHARACTER	1	WQ_ARROW	Arrow Eyecatcher
(3)	CHARACTER	3	WQ_DFH	DFH
(6)	CHARACTER	2	WQ_DOMID	Domain id
(8)	CHARACTER	8	WQ_BLOCK_NAME	Control block name
(10)	ADDRESS	4	WQ_NEXT	-> next in chain
(14)	ADDRESS	4	WQ_PREV	-> prev in chain
(18)	FULLWORD	4	WQ_CALLER_DOMAIN	Index of waiting domain
(1C)	FULLWORD	4	WQ_DOMAIN_TOKEN	Ind of dom waited for or 0
(20)	HALFWORD	2	WQ_PHASE	Phase waited for
(22)	CHARACTER	2	*	Filler
(24)	ADDRESS	4	WQ_SUSP_TOKEN	Suspend token from DS
(28)	CHARACTER		*	Filler

Subpool and Lock Token

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(0)	STRUCTURE	8	SUBPTOK	Subpool token
(0)	ADDRESS	4	SUBPTOK_P	-> subpool token
(4)	FULLWORD	4	SUBPTOK_N	Length subpool token

Constants

Len	**Type**	**Value**	**Name**	**Description**
8	CHARACTER	WQHEAD	WQ_HEAD_BLOCK_NAME	Wait queue head (dummy) name

DMCB4 Domain record

Segment Name = DFHDMCB4
DESCRIPTIVE NAME = CICS/MVS Domain Manager (DM)
Control Blocks 4.
Function =
This file contains data structure
declarations used by the Domain Manager.
The data structures is :
DOMAIN_RECORD - DM CICS Catalog information
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
Domain record

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	30	DOMAIN_RECORD	Domain record
(0)	CHARACTER	16	DR_PREFIX	Domain record prefix area
(0)	HALFWORD	2	DR_LENGTH	Length
(2)	CHARACTER	1	DR_ARROW	Arrow Eyecatcher
(3)	CHARACTER	3	DR_DFH	DFH
(6)	CHARACTER	2	DR_DOMID	Domain id
(8)	CHARACTER	8	DR_BLOCK_NAME	Control block name
(10)	FULLWORD	4	DR_DOMAIN_TOKEN	Domain index
(14)	CHARACTER	8	DR_PROG_NAME	Init program name
(1C)	CHARACTER	2	DR_DOMAIN_ID	Abbrev. domain name

DMENC Domain manager enf state

-

DFHDMENC

DFHDMENC is the copy book that describes the domain manager ENF key 8 state.

-

ENF_ANCHOR

The ENF_ANCHOR control block acts as an anchor for the domain manager event notification facility. This control block is anchored in the domain manager anchor block.

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(0)	STRUCTURE	48	ENF_ANCHOR	
(0)	UNSIGNED	2	ENF_ANCHOR_ LENGTH	
(2)	CHARACTER	14	ENF_ANCHOR_EYE	
(10)	ADDRESS	4	ENF_PUBLIC_QUEUE	
(14)	ADDRESS	4	ENF_PRIVATE_ QUEUE	
(18)	BITSTRING	4	ENF_WAKEUP_ECB	
(18)	BITSTRING	1	*	needed by DSECTGEN
	1...		*	
	.1..		ENF_WAKEUP_ ECB_POSTED	
(1C)	CHARACTER	4	*	reserved
(20)	CHARACTER	16	ENF_EVENT_ARRAY (1)	
(20)	ADDRESS	4	ENF_EVENT_ ARRAY_LISTENER	
(24)	ADDRESS	4	*	
(28)	CHARACTER	8	ENF_EVENT_ ARRAY_TIME	

--
-

ENF_ LISTEN_ELEM

An ENF_ LISTEN_ ELEM is allocated when a domain issues a LISTEN request. The domain index of the domain that is listening is recorded, and the gate index of the gate to be invoked when the event occurs.

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(0)	STRUCTURE	36	ENF_LISTEN_ELEM	
(0)	UNSIGNED	2	ENF_LISTEN_ ELEM_LENGTH	
(2)	CHARACTER	14	ENF_LISTEN_ ELEM_EYE	
(10)	ADDRESS	4	ENF_LISTEN_ ELEM_NEXT	
(14)	UNSIGNED	4	ENF_LISTEN_ ELEM_CODE	
(18)	UNSIGNED	4	ENF_LISTEN_ ELEM_DOMAIN	
(1C)	UNSIGNED	4	ENF_LISTEN_ ELEM_GATE	
(20)	CHARACTER	4	*	
(20)	BITSTRING	1	*	needed by DSECTGEN
	1...		ENF_LISTEN_ ELEM_DELETED	
(21)	BITSTRING	3	*	

```
--
 -

   ENF_ NOTIFY_ELEM

     Notify elements are passed from the ENF SRBEXIT to the ENF
     listening task. ENF notify elements are allocated from CICS key
     subpool 250 storage by the SRB, and are freed by the listening
     task. These elements take the following format
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	ENF_ELEM	
(0)	UNSIGNED	2	ENF_ELEM_LENGTH	
(2)	CHARACTER	14	ENF_ELEM_EYE	
(10)	ADDRESS	4	ENF_ELEM_NEXT	
(10)	ADDRESS	4	ENF_ELEM_ LISTENER	
(14)	UNSIGNED	4	ENF_ELEM_CODE	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	1	NUMBER_OF_ ENF_EVENTS	
4	DECIMAL	2147483647	UNKNOWN_EVENT	

DSANC Dispatcher domain anchor block

```
CONTROL BLOCK NAME = DFHDSANC
DESCRIPTIVE NAME = CICS Dispatcher Anchor Block
FUNCTION =
    This include contains the definition of the Dispatcher
    Anchor Block. It also contains definitions of the DS_TCB,
    Sub_dispatcher, Stimer and Authorised blocks. See below
    for descriptions.
    The anchor block contains all dispatcher-related information
    that is not task, or suspend_resume_area specific.
LIFETIME =
    Dispatcher Lifetime.
STORAGE CLASS =
    OS Getmained from subpool 0.
LOCATION =
    Held by Kernel
INNER CONTROL BLOCKS =
    DS_TCB contains information associated with particular MVS
    TCBs controlled by the Dispatcher. This consists mainly of
    wait related data, eg the wake up ecb for the TCB.
    There is also a macro included here to post the wake up ecb
    of a particular TCB.
    Sub_dispatcher data is associated with one particular mode.
    Currently there is only one TCB per mode, but in case of
    more being introduced, we should distinguish between TCB
    and mode-related data. The key data is concerned with
    the dispatchable chain of tasks with the sub-dispatcher's
    mode.
    The STimer block contains an array of blocks to associate
    with the up to 11 outstanding stimerm calls that can be
    issued by dispatcher.
NOTES :
 DEPENDENCIES = XA
 RESTRICTIONS =
 MODULE TYPE = Control block definition
EXTERNAL REFERENCES = None
  DATA AREAS = None
  CONTROL BLOCKS = None
  GLOBAL VARIABLES (Macro pass) =
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1684	ANCHOR	
(0)	CHARACTER	16	EYE_CATCHER	eye catcher
(0)	HALFWORD	2	CB_LENGTH	Length of cb

Offset Hex	Type	Len	Name (Dim)	Description
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctlblock name
Dispatcher state info				
(10)	CHARACTER	132	DISPATCHER_STATE	
DISPATCHER STATE INFO KEPT IN THE CICS CATALOG				
(10)	HALFWORD	2	NUMBER_ OF_SUBTASKS	No. CO mode TCBS
(12)	UNSIGNED	2	PRIORITY_ MULTIPLIER	Priority Aging factor
(14)	ADDRESS	4	DEAD_DS_TCBS	ds_tcbs whose TCBs have terminated but that can't be freed yet.
(18)	CHARACTER	8	SCAN_DELAY_ INTERVAL	icvtsd
(20)	CHARACTER	8	MAXIMUM_WAIT_INTERVAL	ICV time
Dispatcher state constants set up in DFHDSDM.				
(28)	CHARACTER	8	SO_OFTEN_SHP	checking interval for scan_hand_postables
(30)	CHARACTER	8	PHS1_PERIOD_ LENGTH	
(38)	CHARACTER	8	PHS1_PRIORITY_ BONUS	
(40)	CHARACTER	8	SO_OFTEN_CE	Check_executables checking interval
(48)	CHARACTER	8	TIME_OUT_GAP	period between delayed deadlock timeouts
this is the penalty applied to ALL new tasks				
(50)	CHARACTER	8	NEW_TASK_DELAY	
(58)	ADDRESS	4	SM_ISOLATION_ TOKEN	Subspace isolation token used on switches
(5C)	FULLWORD	4	STORE_SHORT_ POINT	for sos processing
(60)	FULLWORD	4	STORE_CRITICAL_ POINT	for sos processing
(64)	FULLWORD	4	NEW_TASK_ PENALTY	
(68)	HALFWORD	2	SCAN_DELAY_INTERVAL_SIT	icvtsd from SIT
(6A)	HALFWORD	2	*	Reserved
Working Counters and State These fields are set to zero during initialisation They are updated as required during dispatcher operation				
(6C)	FULLWORD	4	NUM_TASKS	Current # of tasks
(70)	FULLWORD	4	PEAK_NUM_TASKS	Peak # of tasks
(74)	FULLWORD	4	CURRENT_STORAGE_FREE	Free storage init (16M)
(78)	FULLWORD	4	STORAGE_ SHORTFALL	store_short_point-above>0
(7C)	CHARACTER	8	NEW_TASK_MINUS	dispatch priority modifier for new tasks
(84)	FULLWORD	4	MAXIMUM_WAIT_INTERVAL_SIT	ICV time from SIT
Bit String state flags The following flags are deliberately separated to avoid clashes when updating the bytes under multiple TCB's				
(88)	CHARACTER	1	*	
	1...		SHUTDOWN_DISPATCHER	1= shutdown
	.1..		QUIESCE_IN_PROGRESS	1= quiesce in prog
(89)	CHARACTER	1	*	
	1...		PERFORM_BEFORE_WAIT_UEXIT	set if required
(8A)	CHARACTER	1	*	
	1...		PERFORM_AFTER_WAIT_UEXIT	set if required
(8B)	CHARACTER	1	*	Reserved
The following flags are set in pre_init				
(8C)	UNSIGNED	4	DS_FLAGS	flag strip
	1...		POST_EXIT_ ENABLED	
	.1..		*	
	..1.		*	
The following flags are set under the QR lock				
	...1		BUILD_WAIT_ LIST	Build QR waitlist if set
	.... 1...		*	
	.... .1..		IN_DISPATCHER_PRE_INIT	Set 'TRUE' when dispatcher pre-initialisation is entered. Set off at end. See DFHDSDM

Offset Hex	Type	Len	Name (Dim)	Description
Lock Words These words are used for compare and swap locking FFFFFFFF = locked, 0000000 = unlocked				
(90)	CHARACTER	4	LOCK_WORDS	lockwords
(90)	UNSIGNED	4	EXECUTABLE_ CHAIN_LOCK	set when scanning the executable chain
AP_INTERFACES Fields used in serviceing the AP domain				
(94)	CHARACTER	4	AP	
(94)	ADDRESS	4	CSA_ADDRESS	Addr of the CICS CSA
ECB queue This chain is for aliens to chain requests to the dispatcher for a service. This is to be used when DFHXMP (in the FOR) wants the AOR to post an AOR ECB. Rather than doing an MVS post (with the overhead of an SRB) it will queue the request so the local dispatcher can do a local post or, even better, do a hand post. This chain will be serviced by DFHDSTCB just before its dispatcher scan.				
(98)	CHARACTER	8	ECB_Q_DW	Double Word for CDS
(98)	ADDRESS	4	Z_ANCHOR	Anchor for ECB Q chain
(9C)	UNSIGNED	4	Z_NUMBER	Number in Queue
Special tasks area. This area keeps track of the special task CSTP. This tasks can issue special WAITs, and we must note when these special requests have been issued.				
(A0)	CHARACTER	12	SPECIAL_AREA	
(A0)	CHARACTER	12	CSTP_AREA	
(A0)	ADDRESS	4	CSTP_TASK_REF	TCP's task block
(A4)	ADDRESS	4	CSTP_ECB_LIST	TCP's ecb list
(A8)	BITSTRING	1	CSTP_FLAGS	TCP's flags
	1...		CSTP_WAITING	TCP's is in special wait
	.1..		CSTP_MUST_DSP	CSATCPEV was set
	..11 1111		*	reserved
(A9)	UNSIGNED	3	*	
The Executable Chain. This chain is a list of all currently attached tasks. It is used to implement the AMAX,Interval,and Timeout scans. Task detach requires that an element be removed from the middle of this chain. Because of this, only one TCB is allowed to browse this chain at a time. If another TCB finds the chain 'locked' it can assume that the tcb that is currently scanning the chain will perform any required operations on the behalf of all tcbs. A Compare and Swap "push" to the top of the chain is always allowed.				
(AC)	CHARACTER	4	EXECUTABLE_CHAIN	
(AC)	ADDRESS	4	EXECUTABLE_ HEADER	
Roots of dispatcher control blocks				
(B0)	CHARACTER	16	TASK_CELL_ROOT	PTR TO TASK BLOCKS
(B0)	ADDRESS	4	PAM_ADDR	Ptr to page alloc map
(B4)	FULLWORD	4	CELL_COUNT	number of cells in pool
(B8)	CHARACTER	8	FREE_CHAIN_CDS	FREE CHAIN HEADER
(B8)	ADDRESS	4	FREE_CHAIN_PTR	PTR TO FIRST FREE CELL
(BC)	UNSIGNED	4	FREE_CHAIN_ COUNT	CDS SAFETY COUNT
(C0)	CHARACTER	16	USER_TASK_ROOT	Ptr to user task blocks
(C0)	ADDRESS	4	PAM_ADDR	
(C4)	FULLWORD	4	CELL_COUNT	
(C8)	CHARACTER	8	FREE_CHAIN_CDS	
(C8)	ADDRESS	4	FREE_CHAIN_PTR	
(CC)	UNSIGNED	4	FREE_CHAIN_ COUNT	
(D0)	CHARACTER	16	SUSPEND_ CELL_ROOT	Ptr to suspend blocks
(D0)	ADDRESS	4	PAM_ADDR	
(D4)	FULLWORD	4	CELL_COUNT	
(D8)	CHARACTER	8	FREE_CHAIN_CDS	
(D8)	ADDRESS	4	FREE_CHAIN_PTR	
(DC)	UNSIGNED	4	FREE_CHAIN_ COUNT	
(E0)	CHARACTER	16	USER_EXTENSION_ ROOT	root of ecb extension blocks
(E0)	ADDRESS	4	PAM_ADDR	
(E4)	FULLWORD	4	CELL_COUNT	
(E8)	CHARACTER	8	FREE_CHAIN_CDS	
(E8)	ADDRESS	4	FREE_CHAIN_PTR	
(EC)	UNSIGNED	4	FREE_CHAIN_ COUNT	
(F0)	CHARACTER	16	EXTENSION_ CELL_ROOT	root of ecb extension blocks
(F0)	ADDRESS	4	PAM_ADDR	
(F4)	FULLWORD	4	CELL_COUNT	
(F8)	CHARACTER	8	FREE_CHAIN_CDS	
(F8)	ADDRESS	4	FREE_CHAIN_PTR	
(FC)	UNSIGNED	4	FREE_CHAIN_ COUNT	

Offset Hex	Type	Len	Name (Dim)	Description
	Hand Postable Chain. Define all fields relating to the anchor portion of the hand postable Q. Tasks on this Q expect that their ECB'S can be posted by an OI of the post bit in the ECB.			
(100)	CHARACTER	24	HAND_POSTABLES	the hand postable q
(100)	ADDRESS	4	HAND_POSTABLE_ CHAIN	Anchor for hpq
(104)	ADDRESS	4	HPT_LAST_PTR	Last entry in HP chain
	The following fields (hpt_wait_list_xxx) describe the wait list used by the quasi-reentrant (QR) TCB when invoking the MVS WAIT during partition exit. The list consists of the wakeup ecb, other special ecbs, and all waiting OLD_WAIT ecbs being waited on by tasks in the handpostable chain			
(108)	ADDRESS	4	HPT_WAIT_ LIST_START	Actual begining of list
(10C)	ADDRESS	4	HPT_WAIT_ LIST_END	First byte "AFTER" the end of the wait list
(110)	ADDRESS	4	HPT_WAIT_ LIST_CURSOR	Ptr -> the next available slot in the wait list
(114)	UNSIGNED	2	HPT_WAIT_ LIST_SIZE	How many ECBs the wait- list will hold.
(116)	UNSIGNED	2	*	Reserved
(118)	CHARACTER	8	DELAY_QUEUE	
	The delay queue consists of tasks which have received a resume request which we wish to delay until either a specified interval has expired, or CICS has nothing better to do. This facility is used by high priority server tasks such as CSNC which do not neccessarily want to be awoken as soon as requests arrive. This allows a CICS server task to achieve batching under the CICS TCB, this method of batching is seperate from that used to reduce the MVS dispatching overhead, the delay queue is intended to offer a mechanism for server tasks to reduce the CICS dispatching overhead.			
(118)	ADDRESS	4	DELAY_QUEUE_ HEAD	head of chain of tasks
(11C)	CHARACTER	4	DELAY_QUEUE_ TIME	earliest delayed work
	TIME Fields			
(120)	CHARACTER	80	TIMER	
(120)	CHARACTER	8	CURRENT_TIME	system time
(128)	CHARACTER	8	NEXT_CE_TIME	Next time the check_ executables routine is due
(130)	CHARACTER	8	NEXT_SHP_TIME	Next time the hand_ postable_scan (quasi-reent function) is due
(138)	CHARACTER	8	NEXT_TIMEOUT_ TIME	Earliest time for deadlock timeout since last timeout
(140)	CHARACTER	8	NEXT_TI_EVENT	Next scheduled event for the timer domain. This is set by the ?DFHTITST macro in DFHDSTCB
(148)	CHARACTER	8	EXPIRATION_ TOKEN	input to DFHTITST macro
(150)	CHARACTER	8	NEXT_TCP_ DISPATCH_TIME	
(158)	CHARACTER	8	DSCSA_WORK	work area for DFHDSCSA
(160)	CHARACTER	8	SAVED_NEXT_ TCP_DISPATCH_TIME	value of next_tcp_dispatch_time while cstp_waiting is off
(168)	UNSIGNED	4	QR_CPU_PERCENT	Percent cpu usage by QR TCB
(16C)	CHARACTER	4	*	Reserved
(170)	CHARACTER	8	PHS1_PRIORITY	
(170)	BITSTRING	4	PHS1_PRIORITY_ HIGH	
(174)	UNSIGNED	4	PHS1_PRIORITY_ LOW	
(178)	CHARACTER	4	KERN_ANCHOR	KE domain anch
(17C)	UNSIGNED	1	NEXT_FREE_SUBD	index of next free sub_disp array element
(17D)	CHARACTER	3	*	reserved

Offset Hex	Type	Len	Name (Dim)	Description

The mode/sub_dispatcher control blocks A SUB_Dispatcher is responsible for a given disp. "mode". It contains a list of dispatcher tcbs owned by this mode, a Dispatchable Q that is a list of tasks that are ready to be dispatched. (ie not suspended) And a set of flags representing the state for this sub dispatcher. In this release, there is exactly one tcb for each sub_disp. The Modes in CICS 3.1.1 are: 1. QR: Quasi-Reentrant. This mode runs all old CICS non-reentrant code. It also runs all application code. RO: Resource Owning Tasks switch to this mode to perform operations that will tie up a TCB for a long period of time. An example open and close files or perform any BLDL operations. Tasks running in this mode run concurrently with any other tasks in the system. CO: Concurrent Mode. Tasks running in this mode run concurrently with any other tasks in the system. Tasks in this mode are expected to give control back to the dispatcher in a reasonable time. CO mode can be viewed as a superior VSAM subtask mode. The current users of CO mode are all the old VSAM subtask users. TSP,FCP,TDP and JCP and domain service tasks, eg for TI and SM. In CICS 3.3 the following TCB was added: SZ: Secondary LU support mode. Tasks running in this mode run concurrently with any other tasks in the system. This mode is used by tasks processing EXEC CICS FEPI requests. This mode is NOT for general purpose use, but is reserved exclusively for use by the secondary LU support code. In CICS 4.1 the following TCB was added: RP: ONC/RPC support mode This mode is used by tasks using the ONC/RPC feature and is intended as a tactical solution only. In CICS 4.2 the following TCB was added: FO: file open/close mode This mode is used rather than RO mode to avoid the possible delay caused to other tasks when migrated files are opened (takes a long time). The sub_dispatchers are implemented as a 6 deep array within the dispatcher anchor block. The array indexes (1..6) correspond with (QR,RO,CO,SZ,RP,FO) modes

Offset Hex	Type	Len	Name (Dim)	Description
(180)	CHARACTER	56	SUB_DISP (20)	Modes in order shown above
(180)	CHARACTER	16	SD_EYE_CATCHER	
(180)	HALFWORD	2	CB_LENGTH	
(182)	CHARACTER	1	ARROW	
(183)	CHARACTER	3	DFH	
(186)	CHARACTER	2	DOMID	
(188)	CHARACTER	8	BLK_NAME	
(190)	CHARACTER	8	BATCH_CONTROL	
(190)	FULLWORD	4	BATCH_SIZE	
(194)	FULLWORD	4	BATCH_CURRENT	
(198)	ADDRESS	4	TCB_LIST	
(19C)	HALFWORD	2	TCB_COUNT	
(19E)	HALFWORD	2	RELATIVE_ PRIORITY	
(1A0)	BITSTRING	4	SUBD_FLAGS	
	1...		MODE_ACTIVE	
	.1..		CHANGE_ MODE_POSSIBLE	
	..1.		EXEC_CAPABLE	
	...1		LE_CICS	
	.... 1...		OPEN_MODE	
	.... .1..		TCBKEY9	
	.... ..1.		INHERIT_SS	
	1		ESSENTIAL_TCB	
(1A1)	1...		MULTIPLE_TCBS	
(1A4)	UNSIGNED	4	SUBD_MODE	
(1A8)	CHARACTER	2	SUBD_MODENAME	
(1AA)	CHARACTER	2	PARENT_MODENAME	
(1AC)	UNSIGNED	4	OPEN_INDEX	
(1B0)	CHARACTER	8	TCB_ID_RANGE	
(1B0)	CHARACTER	1	*	
(1B1)	UNSIGNED	3	NEXT_ID	
(1B4)	CHARACTER	1	*	
(1B5)	UNSIGNED	3	LAST_ID	

Lock for getmains from outside CICS Storage. Using DFHKERN type(lock/unlock)

Offset Hex	Type	Len	Name (Dim)	Description
(5E0)	CHARACTER	8	GETPAGE_LOCK	DFHKERN LOCK FOR GETMAIN

Pointer to the Statistics Record Buffer The stats mapping DSECT is DFHDSGPS. To map this buffer set dfhdsgps_ptr = stats_buffer_ptr.

Offset Hex	Type	Len	Name (Dim)	Description
(5E8)	ADDRESS	4	STATS_BUFFER_PTR	Ptr to Stats Buffer
(5EC)	CHARACTER	4	*	Reserved

Statistics Last Reset Time.

Offset Hex	Type	Len	Name (Dim)	Description
(5F0)	CHARACTER	8	LAST_RESET_TIME	

Miscellaneous Tokens and Pointers

Offset Hex	Type	Len	Name (Dim)	Description
(5F8)	CHARACTER	8	BRTOKEN_SUBPOOL	SUBPOOL FOR BROWSE TOKNS
(600)	CHARACTER	4	DSIT_LOCK_TOKEN	Lock token for dsit

Offset Hex	Type	Len	Name (Dim)	Description
(604)	ADDRESS	4	POST_EXIT_ ADDRESS	Addr of post exit rtn

The following WL table is used to keep track of the average length of the last few MVS WAITs issued under the QR TCB.

Offset Hex	Type	Len	Name (Dim)	Description
(608)	CHARACTER	60	WL	
(608)	CHARACTER	8	WL_AVERAGE_ DURATION	8 byte
(608)	CHARACTER	2	*	
(60A)	FULLWORD	4	WL_AVERAGE	4 byte average
(60E)	CHARACTER	2	*	
(610)	FULLWORD	4	WL_SUM	sum of last WL_N WAITs
(614)	FULLWORD	4	WL_N	number of table entries
(618)	ADDRESS	4	WL_OLDEST	oldest entry
(61C)	ADDRESS	4	WL_FIRST	first entry
(620)	ADDRESS	4	WL_LAST	last entry
(624)	FULLWORD	4	WL_DURATION (8)	the entries

The following fields are used to manage open TCBs.

Offset Hex	Type	Len	Name (Dim)	Description
(644)	CHARACTER	76	OPEN_TCBS	
(644)	ADDRESS	4	OPEN_TCB_ MANAGEMENT_LOCK	
(648)	CHARACTER	24	COUNTS	
(648)	UNSIGNED	4	CURR_ALLOC_ OPEN_TCBS	TCBs allocated to current tasks
(64C)	UNSIGNED	4	HIGH_ALLOC_ OPEN_TCBS	highwater mark for CURR_ALLOC_OPEN_TCBS
(650)	UNSIGNED	4	CURR_OPEN_TCBS	total no. open TCBs currently in existence
(654)	UNSIGNED	4	HIGH_OPEN_TCBS	highwater mark for CURR_OPEN_TCBS
(658)	UNSIGNED	4	MAXOPENTCBS	SIT/override limiting no. of open TCBs
(65C)	UNSIGNED	4	SUSPENDED_ AWAITING_OPEN_TCB	no. tasks suspended awaiting open TCBs
(660)	BITSTRING	4	OPEN_FLAGS	
	1...		TRANISO	on if TRANISO = YES
	.1..		LOCK_FAILED	open mgmt lock has failed
	..1.		ALREADY_ AT_MAXOPEN	at MAXOPENTCB
(660)	BITSTRING	3	*	reserved
(664)	CHARACTER	8	SM_VARIABLE_ SUBPOOL_TOKEN	hash tbl sbpl@LCA
(66C)	ADDRESS	4	AWAITING_ OPEN_TCB	chain of tasks awaiting a free TCB
(670)	ADDRESS	4	AWAITING_ OPEN_TCB_END	end of chain of tasks awaiting a free TCB
(674)	CHARACTER	4	*	reserved for open tcbs
(678)	CHARACTER	24	FREE_CHAINS	Arrays indexed by open TCB type
(678)	ADDRESS	4	FREE_OPEN_ BASESPACE_DS_TCBS (3)	chain of basespace TCBs unalloc'd to tasks
(684)	ADDRESS	4	FREE_OPEN_ SUBSPACE_DS_TCBS (3)	hash chns of subsp TCBs unalloc'd to tasks
(690)	ADDRESS	4	FREE_DS_TCBS	chain of free ds_tcbs

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	224	DS_TCB	

next_ dead_ds-tcb changes, dfhdsani must be changed.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER	24	DS_TCB_PART1	
(0)	CHARACTER	16	EYE_CATCHER	
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctlblock name
(10)	ADDRESS	4	NEXT_TCB	ptr to next tcb ctl block Last one is set to X'00'
(14)	ADDRESS	4	TCB_SUBD_PTR	Ptr to owning subdisp cb
(18)	CHARACTER	16	DS_TCB_PART2	

(for CDS and CS reasons).

Offset Hex	Type	Len	Name (Dim)	Description
(18)	UNSIGNED	4	INSTANCE_COUNT	TCB instance
(18)	BITSTRING	3	*	
(1B)	1		TCB_AVAILABLE	1 = TCB still active
(1C)	CHARACTER	8	DISPATCHABLE_ CHAIN	the dispatchable q
(1C)	ADDRESS	4	FRONT_PTR	
(20)	ADDRESS	4	BACK_PTR	
(24)	CHARACTER	4	KE_TASK_TOKEN	TASK_TOKEN passed back by DFHKEDS CREATE_TCB
(28)	CHARACTER	184	DS_TCB_PART3	

Offset Hex	Type	Len	Name (Dim)	Description
(28)	UNSIGNED	4	WAKE_UP_ECB	ECB used to wake TCB
	1...		TCB_WAITING	waiting bit.
	.1..		TCB_POSTED	used for tcb_state
(2C)	ADDRESS	4	RUNNING_TASK	Currently running task
(30)	ADDRESS	4	TCB_ANC_ADDR	Ptr -> Anchor Block
(34)	CHARACTER	4	*	give first_timeout correct alignment
(38)	CHARACTER	8	FIRST_TIMEOUT	first time task due to timeout on this TCB
(40)	ADDRESS	4	STIMER_ BLOCK_PTR	Address of stimer block
(44)	CHARACTER	8	TCB_SUBD_NAME	QR|RO|CO|SZ|RP|FO
(4C)	UNSIGNED	1	TCB_MODE	As per dsat modes 1 = Qr mode 2 = RO mode 3 = CO mode 4 = SZ Mode 5 = RP mode 6 = FO mode
(4D)	BITSTRING	1	DS_TCB_FLAGS	
	1...		PERFORM_KE_READ_TIME	KE_READ_TIME needed
	.1..		DELETE_ TCB_RECEIVED	delete_tcb request
	..1.		ESSENTIAL_TCB	essential_tcb(yes)
(4E)	BITSTRING	1	DS_TCB_FLAGS2	
	1...		SHUTDOWN_TCB	NB needs its own byte
(4F)	CHARACTER	1	*	reserved
(50)	CHARACTER	8	WAIT_FINISH	STCK when Ptn exit starts
(58)	CHARACTER	8	WAIT_START	STCK when Ptn exit completes
(60)	CHARACTER	8	ANC_TCB_ WAIT_TIME	OP System wait time
(68)	CHARACTER	8	ANC_TCB_ DISP_TIME	TCB dispatch time
(70)	FULLWORD	4	ANC_SYSTEM_ WAITS	No of partition exits
(74)	FULLWORD	4	*	Reserved

The following fields are used to manage open TCBs

Offset Hex	Type	Len	Name (Dim)	Description
(78)	CHARACTER	36	OPEN_DS_ TCB_STATE	Fields for open TCBs
(78)	CHARACTER	8	MOST_RECENT_ USE	last time TCB used
(80)	ADDRESS	4	SUBSPACE_TOKEN	TCB's associated subsp
(84)	ADDRESS	4	OWNING_TASK	Task owning this TCB
(88)	ADDRESS	4	NEXT_OPEN_FREE	Open TCB chain fwd ptr
(8C)	BITSTRING	4	OPEN_FLAGS	
	1...		SUBSPACE_ ELIGIBLE	1 = TCB attached with subspace
	.1..		OPEN_MODE	open(open_yes) mode
	..1.		DELETE_ TCB_ISSUED	set before issuing DELETE_TCB
	...1		TCB_TERM_BEFORE_DELETE_TCB	TCB terminated before DELETE_TCB issued (implies TCB terminated catastrophically)
(8C)	BITSTRING	3	*	reserved
(90)	CHARACTER	12	*	reserved for open TCBs
(9C)	CHARACTER	5	TCB_ID	for trace entries
(9C)	CHARACTER	2	TCB_MODENAME	modename
(9E)	UNSIGNED	3	TCB_NUMBER	alphanumeric number
(A1)	CHARACTER	3	*	Reserved
(A4)	ADDRESS	4	*	Reserved

'Saved' statistical values used in the calculation of CPU utilisation.

Offset Hex	Type	Len	Name (Dim)	Description
(A8)	CHARACTER	16	TCB_SAVED_ CPU_FIELDS	
(A8)	CHARACTER	8	TCB_SAVE_ WAIT_TIME	
(B0)	CHARACTER	8	TCB_SAVE_ ACC_TIME	

The following two fields are used in the calculation of DSGACT, which is the CPU time used by any TCB during a given Statistics Interval. TCB_ TOTAL_ACC_ CPU_TIME is the total CPU time burnt by a TCB TCB_ OLD_CPU_ TIME is the total CPU time burnt by a TCB up to the start of a given Statistics Interval.

Offset Hex	Type	Len	Name (Dim)	Description
(B8)	CHARACTER	8	TCB_TOTAL_ACC_CPU_TIME	
(C0)	CHARACTER	8	TCB_OLD_ CPU_TIME	

The following two fields are used in the calculation of DSGTCT, which is the CPU time used by any TCB whilst processing the DS task during a given Statistics Interval. TCB_ DS_TOT_ ACC_CPU_TIME is the total CPU time burnt by a TCB whilst executing the DS task. TCB_ DS_OLD_ CPU_TIME is the total CPU time burnt by a TCB up to the start of a given Statistics Interval.

Offset Hex	Type	Len	Name (Dim)	Description
(C8)	CHARACTER	8	TCB_DS_TOT_ACC_CPU_TIME	
(D0)	CHARACTER	8	TCB_DS_ OLD_CPU_TIME	

dfhdsani must be changed.

Offset Hex	Type	Len	Name (Dim)	Description
(D8)	UNSIGNED	4	ESTAE_WAITERS_ ECB	for ESTAE exit WAITs
(DC)	ADDRESS	4	NEXT_DEAD_ DS_TCB	chain of ESTAE waiters

Sub_dispatcher
The subdispatcher control block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	SUB_DISPATCHER	Subdispatcher Control blk
(0)	CHARACTER	16	SD_EYE_CATCHER	eye catcher
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctlblock name
(10)	CHARACTER	8	BATCH_CONTROL	
(10)	FULLWORD	4	BATCH_SIZE	total batch size
(14)	FULLWORD	4	BATCH_CURRENT	no reqs left to fill batch
(18)	ADDRESS	4	TCB_LIST	Ptr to a list of tcb's owned by this mode.
(1C)	HALFWORD	2	TCB_COUNT	TCBs for this mode
(1E)	HALFWORD	2	RELATIVE_ PRIORITY	prty relative to QR
(20)	BITSTRING	4	SUBD_FLAGS	Flags word
	1...		MODE_ACTIVE	A successful activate_mode has been issued.
	.1..		CHANGE_ MODE_POSSIBLE	At least one TCB exists for this mode
	..1.		EXEC_CAPABLE	This mode supports EXEC CICS commands and LE.
	...1		LE_CICS	On - LE will use CICS services, off - LE will use MVS services
	.... 1...		OPEN_MODE	1 = open(yes) specified on activate_mode
	.... .1..		TCBKEY9	1 = key 9 TCBs
	.... ..1.		INHERIT_SS	1 = inherits subspace
	1		ESSENTIAL_TCB	1 = terminate CICS if this TCB fails and can't recover
(21)	1...		MULTIPLE_TCBS	1 = more than one TCB allowed for this mode
(24)	UNSIGNED	4	SUBD_MODE	Default mode
(28)	CHARACTER	2	SUBD_MODENAME	from activate_mode
(2A)	CHARACTER	2	PARENT_MODENAME	mode of TCB used to ATTACH TCBs in this mode
(2C)	UNSIGNED	4	OPEN_INDEX	index into array of..
open TCB types (0 if not open)				
(30)	CHARACTER	8	TCB_ID_RANGE	current range of available tcb ids for this mode.
(30)	CHARACTER	1	*	reserved
(31)	UNSIGNED	3	NEXT_ID	next available value in current range
(34)	CHARACTER	1	*	reserved
(35)	UNSIGNED	3	LAST_ID	highest available value

Double Chains.
A Double Chain is a type of linked list that is designed to provide a sorted list of tasks whilst allowing concurrent push/pop operations on it from multiple TCBS..
It consists of 2 linked lists. These are described as the "front" and the "back" halves of the Q.
Any TCB can "push" a new element onto the "Front" half with a Compare and Swap instruction.
When a TCB wants to pop a task of the Q, it "hides" the frontq by zeroing the frontq ptr. Any future pushes to the front half therefor start a fresh front half.
The TCB then sorts and merges the tasks from the hidden front half down onto the back half.
The back half then consists of a list of tasks sorted in priority Order.
The Dispatchable chain is implemented as a double chain.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	DOUBLE_CHAIN	
(0)	ADDRESS	4	FRONT_PTR	Publicly appendable half
(4)	ADDRESS	4	BACK_PTR	Hidden/sorted half

Stimer Block
The block of storage needed for the STIMER times and tokens

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	296	STIMER_BLOCK	
(0)	CHARACTER	16	SB_EYE_CATCHER	eye catcher
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character

Offset Hex	Type	Len	Name (Dim)	Description
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctlblock name
(10)	CHARACTER	8	FIRST_STIMER	1st stimer due to expire
(18)	CHARACTER	1	STIMER_FLAGS	Various flags
	1...		STIMER_RUN	Stimer exit has run since last partition exit.
	.111 1111		*	reserved
(19)	UNSIGNED	1	*	
(1A)	UNSIGNED	2	BACKSTOP_ STIMER_INDEX	indicates backstop stimer
(1C)	ADDRESS	4	STIMER_DSTCB	ds_tcb address
(20)	CHARACTER	24	STIMER_ARRAY (0 10)	
(20)	ADDRESS	4	ANCHOR_ADDR	ADDRESS OF ANCHOR BLOCK
(24)	CHARACTER	8	STIMER_TIME	TIME STIMERS DUE TO EXPIRE
(2C)	CHARACTER	4	STIMER_TOKEN	ASSOCIATED TOKENS FROM XA
(30)	ADDRESS	4	STIMER_ ENTRY_ADDR	-> own array element
(34)	ADDRESS	4	STIMER_ BLOCK_ADDR	-> parent stimer block

DSAUSB. This is the address-space-wide (ie. global) dispatcher authorized block. It is key 0, job-step local, and is addressed by the CICS AFCS.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	164	DSAUSB	
(0)	CHARACTER	16	DSSEYECATCH	standard eyecatcher
(0)	HALFWORD	2	CB_LENGTH	
(2)	CHARACTER	1	ARROW	
(3)	CHARACTER	3	DFH	
(6)	CHARACTER	2	DOMID	
(8)	CHARACTER	8	BLK_NAME	
(10)	ADDRESS	4	DSPXENT (0 7)	POST exit entry pts in DSAUT
(30)	ADDRESS	4	DSPXADD (0 7)	POST exit initial entry pts (in POST exit stubs in LPA)
(50)	CHARACTER	72	DSSREGSAV	savearea
(98)	FULLWORD	4	DSPSWAP	DONTSWAP count
(9C)	1...		DSPXENAB	bitstrip giving postexit enable/disable states
(A4)	CHARACTER		DSAUSB_END	end of ctl blk

DSAUTB. This is the TCB-local dispatcher authorized block. It is key 0, TCB-related lsqa, and is addressed by the CICS AFCB.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	104	DSAUTB	
(0)	CHARACTER	16	DSTEYECATCH	standard eyecatcher
(0)	HALFWORD	2	CB_LENGTH	
(2)	CHARACTER	1	ARROW	
(3)	CHARACTER	3	DFH	
(6)	CHARACTER	2	DOMID	
(8)	CHARACTER	8	BLK_NAME	
(10)	ADDRESS	4	DST_DS_TCB_ADDR	addr of this TCB's DS_TCB
(14)	CHARACTER	72	DSTREGSAV	savearea
(5C)	ADDRESS	4	DSTPEXAD	temp for post exit addr
(60)	CHARACTER	8	DSTUSER_PARM	area to hold user parms
(60)	FULLWORD	4	REQUEST_TYPE	caller's request type - hold here for integrity
(64)	FULLWORD	4	PEX_NUM	caller's postexit num - hold here for integrity
(68)	CHARACTER		DSAUTB_END	end of ctl blk

Quickcell Page Allocation Maps.
The dispatcher quickcell mechanisms use page allocation maps to implement the mapping from the cell tokens to the cell addresses.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	DS_CELL_PAM	
(0)	CHARACTER	16	EYE_CATCHER	eye catcher

Offset Hex	Type	Len	Name (Dim)	Description
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctlblock name
(10)	ADDRESS	4	CELL_PAGE_MAP (*)	Array of page addresses

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	2064	DS_TASK_PAM	
(0)	CHARACTER	16	EYE_CATCHER	eye catcher
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctlblock name
(10)	ADDRESS	4	TASK_PAGE_MAP (0 511)	Array of page addresses

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1040	DS_SUSPEND_PAM	
(0)	CHARACTER	16	EYE_CATCHER	eye catcher
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctlblock name
(10)	ADDRESS	4	SUSPEND_PAGE_MAP (0 255)	Array of page addresses

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1040	DS_EXTENSION_PAM	
(0)	CHARACTER	16	EYE_CATCHER	eye catcher
(0)	HALFWORD	2	CB_LENGTH	Length of cb
(2)	CHARACTER	1	ARROW	> character
(3)	CHARACTER	3	DFH	DFH characters
(6)	CHARACTER	2	DOMID	DS for Dispatcher domain
(8)	CHARACTER	8	BLK_NAME	set to ctlblock name
(10)	ADDRESS	4	EXTENSION_ PAGE_MAP (0 255)	Array of page addresses

Constants

Len	Type	Value	Name	Description
4	DECIMAL	0	UNEX_OK	
4	DECIMAL	4	UNEX_NOT_EXTENDED	

DSTBA Task browse area

```
CONTROL BLOCK NAME = DFHDSTBA
DESCRIPTIVE NAME = CICS Dispatcher task browse area
FUNCTION =
     This block indicates where a browse of the CICS tasks should
     resume. The block and task-within-block numbers are used
     to identify where in the chain of task pages we have reached
LIFETIME =
     Dispatcher Browse lifetime
STORAGE CLASS =
     Dispatcher Browse Subpool
LOCATION =
     Pointed to by Browse Token
INNER CONTROL BLOCKS = None
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS =
 MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
  DATA AREAS =
  CONTROL BLOCKS =
  GLOBAL VARIABLES (Macro pass) =
 Task Browse Area for dispatcher browse
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	18	BROWSE_AREA	
(0)	CHARACTER	16	CELL_HEADER	Header
(0)	HALFWORD	2	LEN	Length of browse area
(2)	CHARACTER	1	ARROW	>
(3)	CHARACTER	13	NAME	DFHDSBROWSE
(10)	UNSIGNED	2	CELL_ID	1st half of token of next task *

DSTSK Dispatcher domain task description

```
CONTROL BLOCK NAME = DFHDSTSK
DESCRIPTIVE NAME = CICS Dispatcher Task Area
FUNCTION =
    The Task is the main control block associated with a CICS-
    dispatchable unit by the Dispatcher.
LIFETIME =
    ATTACH (DFHDSAT) to DETACH (DFHDSTCB after return from PUSH)
    Note TASKs are never freed by the Dispatcher but are instead
    managed by the DS quickcell routines.
STORAGE CLASS =
    MVS Subpool 0.
LOCATION =
    Chained off the DS Anchor on various TASKS Chains depending
    on State.
INNER CONTROL BLOCKS =
    EXTENSION. MVS ECB EXTENSION for WAIT_MVSs done by this task
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS =
 MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
  DATA AREAS =
  CONTROL BLOCKS =
  GLOBAL VARIABLES (Macro pass) =
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	256	TASK	
DTA - Dispatcher Task Area				
The default suspend/resume area for a task is imbedded within the task. By placing the suspend/resume area at the start of the task the standard_ cell_fields for both the task and the suspend resume area will be at the start of dsect.				
(0)	CHARACTER	44	DEFAULT_ SUSPRES_AREA	
(0)	CHARACTER	8	CELL_HEADER	
(0)	ADDRESS	4	UNUSED_PTR	
(4)	UNSIGNED	4	CELL_TOKEN	
(4)	UNSIGNED	2	CELL_ID	
(6)	UNSIGNED	2	USE_COUNT	
(8)	ADDRESS	4	STASK	
(C)	CHARACTER	16	RESOURCE_NAME	
(1C)	CHARACTER	8	RESOURCE_TYPE	
(24)	UNSIGNED	1	COMPLETION_CODE	
(25)	UNSIGNED	1	PURGE_TYPE	
(26)	CHARACTER	2	*	
(28)	UNSIGNED	4	SUSPEND_CS_WORD	
(28)	UNSIGNED	1	STATE	
(29)	CHARACTER	3	*	
The data at the start of the DTA is referenced in the dispatcher scans, and may be referenced not just when dispatching the DTA for this task, but also when considering dispatching other tasks.				

Offset Hex	Type	Len	Name (Dim)	Description

Chaining fields for task
There are many chains within the dispatcher, but only 2
chaining fields are required.
The following shows which Chains are Mutualy Exclusive.
Unused or Executable
If a task is on the Executable chain, it can also be on
ONE of the following chains.
Dispatchable(s) (one per TCB)
Hand_ postable
Executable chain = This is the list of all DS tasks.
This chain is used by functions such
as Timeout, that are
interested in scanning sets of tasks
rather than just selecting a task from
the front of a list.
Note that a task can be on other chains
as well as this one.
GENERAL_ CHAIN = This is a chain field used for the
following chains.
1. Free - Alias the 'Unused',or
the 'Not in use'
chain. All spellings
are talking about the
same thing
The next chain is dealing
with tasks that are
' ready '. Ie they are
not suspended or waiting.
2. Dispatchable. - The List(s) of
tasks that are waiting to
be dispatched.
3. Hand_ postable. - Tasks are put
here when they issue a
WAIT_ OLDW or a WAIT_ OLDC.
The chain is scan to see
if any ecbs for these tasks
have been 'Hand Posted' by
some program setting the
post bit on in the ECB.
All these fields are just straight forward ptrs to the next
task in the chain.

(2C)	ADDRESS	4	EXECUTABLE_NEXT	
(30)	ADDRESS	4	GENERAL_NEXT	
(34)	ADDRESS	4	HAND_POST_NEXT	
(38)	BITSTRING	8	TIMEOUT_TIME	0 or timeout expiry time in STCK units
(40)	BITSTRING	4	CHAIN_FLAGS	
(40)	BITSTRING	1	CHAIN_FLAGS1	
	1...		HAND_POST_ IGNORE	ignore during hand_postable scan, this task logically removed from hand_postable chain.
	.1..		TEMP_HIGH_ PRIORITY	If this is set to YES give task temporary high priority boost on wakeup. Introduced to give LG defer task a boost on timer pop to stop it getting held up by normal traffic due to its potentially low priority.
	..11 1111		*	
(41)	BITSTRING	1	CHAIN_FLAGS2	Reserved
(42)	BITSTRING	1	CHAIN_FLAGS3	Reserved
(43)	BITSTRING	1	CHAIN_FLAGS4	Reserved

The data in the middle of the DTA is typically referenced
each time this task is dispatched, or made dispatchable.
This data is not usually referenced unless this task is
dispatched, or about to be dispatched.

State related fields that must be compared and swapped together

(44)	UNSIGNED	4	CS_GROUP	
(44)	UNSIGNED	1	TASK_STATE	
(45)	UNSIGNED	1	PURGE_STATUS	
(46)	CHARACTER	2	*	

STCK fields must be on dword boundaries

(48)	BITSTRING	8	DISPATCH_ PRIORITY	sort field for dispatch chains measured in store clock units *
(48)	CHARACTER	7	*	
(4F)	UNSIGNED	1	DISPATCH_ PRIORITY_BIN	bin(8) if prtyage=0
(50)	BITSTRING	8	ENQUEUE_TIME	

Time task was set to particular stat measured in store clock units

(50)	UNSIGNED	4	ENQUEUE_ TIME_IN_SECS	
(58)	BITSTRING	8	PHS1_EXPIRY_TIME	PHS1 expiry time as STCK

Pointers to related blocks

(60)	ADDRESS	4	EXTENSION_ ADDRESS	addr of ds extension cell *
(64)	CHARACTER	4	KERNEL_TASKID	

Offset Hex	Type	Len	Name (Dim)	Description
DFHDSATI inline macro.				
(68)	BITSTRING	1	TASK_MODE	TCB Affinity
1-QR 2-RO 3-CO				
(69)	UNSIGNED	1	TYPE	System | Non_System
1 System 2 Non_System System tasks are not subject to new task penalties.				
(6A)	BITSTRING	1	TASK_MISC_FLAGS	odds and ends
	1...		SPECIAL_TYPE	special task
	.1..		SPECIAL_ TYPE_SMSY	SM special task SMSY
	..1.		SPECIAL_ TYPE_IMMEDIATE_ SHUTDOWN	immediate shutdown task
	...1		PURGEABLE	Does user expect purges?
	.... 1...		BATCH_REQD	Should TCB posts be patched? *
	.... .1..		DELAY_ACTIVE	delay task resumed ?
	.... ..1.		RETRY_REQUEST	continuation of old req
	1		DELAY_OVER_WAIT	allow delay to cross partition exits
(6B)	UNSIGNED	1	PRIORITY	User Assigned Priority high=important *
Data associated with Suspend/Wait				
(6C)	ADDRESS	4	WAIT_TOKEN	Not waiting/suspended if this is 0. May contain ECBADDR, Suspend_token add ETC.
(70)	ADDRESS	4	ECBPARM	ECB or ECBLIST parm to WAIT
(74)	UNSIGNED	1	WAIT_TYPE	Type of WAIT,SUSPEND
1-OLDC 2-MVS 3-OLDW 4-SUSPEND				
(75)	UNSIGNED	1	ECBPARM_TYPE	indicates LIST or SINGLE *
1-SINGLE 2-LIST				
(76)	UNSIGNED	1	TIMEOUT_TYPE	interval/deadlock
Data for communication with TCB task				
(77)	UNSIGNED	1	CURRENT_REQUEST	Current processing to be completed by TCB level code *
(78)	ADDRESS	4	CURRENT_TCB_DATA	pointer to TCB's DS data block
(7C)	ADDRESS	4	CURRENT_ PARM_LIST	pointer to domain call format
(80)	CHARACTER		MIDDLE_END	end of this section of DTA
The data at the end of the DTA is typically referenced infrequently, for example when a task is created or destroyed. Data should not be placed in this section of the DTA if it is referenced on every dispatch of the task.				
(80)	CHARACTER	8	DTA_XM_TXN	XM domain transaction token
(88)	BITSTRING	8	RETRY_SUSPEND_ START	time of last RETRYABLE suspend
(88)	UNSIGNED	4	RETRY_SUSPEND_ START_IN_SECS	
(90)	BITSTRING	8	PRIORITY_ TIME_FACTOR	priority part of above
(98)	CHARACTER	8	DELAY_EXPIRED_ TIME	time con dsptch
(A0)	ADDRESS	4	*	reserved
(A4)	CHARACTER	4	DOMAIN_OWNER	Attaching Domain
(A8)	CHARACTER	4	REPLY_GATE	TASK_REPLY gate in OWNER for this task *
(AC)	CHARACTER	4	USER_TOKEN	Attachers name for task eg XM's TQE *
(B0)	BITSTRING	8	DTIMOUT	Deadlock timeout period for task in Store Clock units
The following fields are used to manage open TCBs				
(B8)	CHARACTER	36	OPEN_TCBS	
(B8)	BITSTRING	4	TYPES_USED	BITS 1 to 32: bit 33-n set if task used nth open type in... OPEN_DS_TCB array (above)
(BC)	ADDRESS	4	AWAITING_ OPEN_TCB_TOKEN	SUSPEND token assoc'd with AWAITING_OPEN_TCB chain
(C0)	BITSTRING	1	OPEN_FLAGS	Flag byte
	1...		UNCLEAN	=1 if task set unclean
	.1..		ADD_SUSPEND_ ISSUED	for await tcb queue
	..11 1111		*	reserved
(C1)	CHARACTER	3	*	reserved for open TCBs
(C4)	ADDRESS	4	AWAIT_CHAIN_FWD	await tcb queue - fwd ptr
(C8)	UNSIGNED	4	NUM_OWNED_ OPEN_TCBS	count of tsk's open TCBs@LBA
(CC)	CHARACTER	4	*	reserved for open TCBs
(D0)	ADDRESS	4	OPEN_DS_TCB (3)	For each open TCB type: addr of task's open TCB
(DC)	BITSTRING	4	ABTERM_ PENDING_ECB	
Wait for ABTERM to end.				
(E0)	BITSTRING	1	GENERAL_FLAGS	
	1...		PULLED_ AND_RECOVERY_SET	Task was "pulled" from a non essential TCB that suffered a non-recoverable error. The task was the subject of a dfhkern recovery_set during the pull processing.
(E1)	CHARACTER		TASK_END	
(E1)	CHARACTER	31	*	

Suspend Resume:- Area Corresponding to a Suspend Token.
Area. :-
SUSPEND_ RESUME_ AREA can have states of RESET|SUSPENDED|RESUMED
UNUSED or PURGED

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	44	SUSPEND_ RESUME_AREA	
(0)	CHARACTER	8	CELL_HEADER	
(0)	ADDRESS	4	UNUSED_PTR	
(4)	UNSIGNED	4	CELL_TOKEN	
(4)	UNSIGNED	2	CELL_ID	
(6)	UNSIGNED	2	USE_COUNT	
	cell chaining fields, token etc			
(8)	ADDRESS	4	STASK	Set when token is suspended
(C)	CHARACTER	16	RESOURCE_NAME	Res. name passed by caller
(1C)	CHARACTER	8	RESOURCE_TYPE	Res. type passed by caller
(24)	UNSIGNED	1	COMPLETION_CODE	Comp code from user
(25)	UNSIGNED	1	PURGE_TYPE	Why was task purged?
(26)	CHARACTER	2	*	
(28)	UNSIGNED	4	SUSPEND_CS_WORD	*
(28)	UNSIGNED	1	STATE	state of S/R area *
(29)	CHARACTER	3	*	

ECB extension. This block is pointed by the task (field
EXTENSION_ ADDRESS).

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	EXTENSION	ecb extension
(0)	CHARACTER	8	CELL_FIELDS	quickcell management fields
(0)	ADDRESS	4	UNUSED_PTR	
(4)	UNSIGNED	4	CELL_TOKEN	
(4)	UNSIGNED	2	CELL_ID	
(6)	UNSIGNED	2	USE_COUNT	
(8)	CHARACTER	24	MVS_EXTENSION	actual ecb extension
(8)	UNSIGNED	1	EXT_VALUE	ECB extension VALUE byte
(9)	BITSTRING	1	EXT_MODE	ECB extension MODE byte
(A)	BITSTRING	2	EXT_RES	ECB extension RESERVED field *
(C)	ADDRESS	4	EXT_POSTEXIT	ECB extension POST EXIT addr *
(10)	CHARACTER	12	EXT_USER	ECB extension user area
(10)	ADDRESS	4	EXT_THISTASK	ECB extension owning task addr *
(14)	UNSIGNED	4	EXT_STATUS	ECB extension status - see below for values *
	The POST routine DFHDSCPX relies on the following field EXT_CHEAPEXIT being at offset X'10' in this control block DO NOT CHANGE IT			
(18)	UNSIGNED	4	EXT_CHEAPEXIT	Addr of CHEAP POST EXIT
(1C)	UNSIGNED	4	*	Reserved

Constants

Len	Type	Value	Name	Description
4	HEX	003E8000	PRI_ALLIGN	
	Enumerated Data types for Task fields TYPE_ OF_TASK is SYSTEM|NON_ SYSTEM			
1	DECIMAL	1	SYSTEM	
1	DECIMAL	2	NON_SYSTEM	
	TIMEOUT_ TYPE IS INTERVAL|DEADLOCK_ DELAYED|DEADLOCK_IMMEDIATE			
1	DECIMAL	1	INTERVAL	
1	DECIMAL	2	DEADLOCK_DELAYED	
1	DECIMAL	3	DEADLOCK_IMMEDIATE	
	PURGE_ STATUS is OK|PURGE_ PENDING|PURGED			
1	DECIMAL	1	PURGE_PENDING	
1	DECIMAL	171	ABTERM_PENDING	
	WAIT_ TYPE is OLDC|MVS|OLDW|SUSPEND			
1	DECIMAL	1	OLDC	
1	DECIMAL	2	MVS	
1	DECIMAL	3	OLDW	
1	DECIMAL	4	SUSPEND	

Len	Type	Value	Name	Description
ECB_ TYPE is LIST|SINGLE				
1	DECIMAL	1	ECB_SINGLE	
1	DECIMAL	2	ECB_LIST	
TASK_ STATE is UNUSED|NON_ EXECUTABLE|DISPATCHABLE|RUNNING_ ABTERM_ ALLOWED|RUNNING_ABTERM_NOT_ALLOWED|SUSPENDED|RESUMED|RESUMED_ EARLY				
1	DECIMAL	2	RUNNING_ABTERM_NOT_ALLOWED	
1	DECIMAL	3	DISPATCHABLE	
1	DECIMAL	4	RUNNING_ABTERM_ALLOWED	
1	DECIMAL	5	RESUMED_EARLY	
CURRENT_ REQUEST IS DETACH SLEEP OR REQUEUE.				
1	DECIMAL	1	DETACH	
1	DECIMAL	2	SLEEP	
1	DECIMAL	3	REQUEUE	
TASKS_ IN_BLOCK is the number of tasks that fit in a page of storage				
4	DECIMAL	15	TASKS_IN_BLOCK	
4	DECIMAL	82	SUSPEND_RESUME_AREAS_IN_BLOCK	*
The following constants describe the values taken by the ecb extension status field, EXT_STATUS. Note that the field is changed via Compare-and-swap				
4	DECIMAL	0	EXT_ST_UNUSED	Unused
4	DECIMAL	1	EXT_ST_EXTEND	Started to extd ecbs
4	DECIMAL	2	EXT_ST_EXIT_RAN	POSTEXIT ran before extending complete
4	DECIMAL	3	EXT_ST_EXT_COMPL	Extending complete
EXTENSIONS_IN_BLOCK = number of exts that fit in a page of storage				
4	DECIMAL	124	EXTENSIONS_ IN_BLOCK	

DTCPS Data tables local access anchor blocks

DTCHD_BLOCK, the Data Tables Connect Header Block, is allocated once per region which has performed client initialization processing to allow connections to other regions. It is addressed via the region anchor. It contains information used by the supervisor routines which establish and validate connections to files associated with data tables in server regions.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	544	DTCHD_BLOCK	DT Connect Header block
(0)	CHARACTER	16	DTCHD_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTCHD_LEN	Length of connect anchor
(2)	CHARACTER	1	DTCHD_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTCHD_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTCHD_ID	Eye catcher 'CONNECT'
(10)	CHARACTER	8	DTCHD_VECTOR_ DESC	Connect vector descriptor
(10)	ADDRESS	4	DTCHD_VECTOR_ PTR	Address of connect vector
(14)	FULLWORD	4	DTCHD_VECTOR_ SIZE	Total connect vector entries
(18)	FULLWORD	4	DTCHD_VECTOR_HI_ACTIVE_INDEX	Highest index for which current DTCON_COUNT is non-zero - never less than true value but might be more
(1C)	ADDRESS	4	DTCHD_CALLER_RB	Address of RB which issued initialization call, checked against RB issuing CONNECT, DISCONNECT or record retrieval requests
(20)	BITSTRING	512	DTCHD_LX_MAP	Bit map indexed by LX 0-4095 indicating whether ETCON has been performed for a server region using that LX value

DTCON_VECTOR, the Data Tables Connect Vector, is effectively a variable length extension of the Connect Header Block, but it is stored separately to allow it to be reallocated at a larger size if necessary. It contains information used to establish and validate cross-memory connections to data tables.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	DTCON_VECTOR (*)	Data Tables Connect Vector
(0)	FULLWORD	4	DTCON_COUNT	Number of valid connections to the remote file instance identified by this entry
(4)	UNSIGNED	2	DTCON_ASID	Target address space id - for diagnostic purposes only
(6)	CHARACTER	10	DTCON_INFO	Coded connection information which is used for retrieval
(6)	UNSIGNED	2	DTCON_LX	PC linkage index
(8)	UNSIGNED	4	DTCON_FILE_ REUSE	Server file reuse counter
(C)	ADDRESS	4	DTCON_FILE_ TOKEN	Server file block address
(10)	CHARACTER	8	DTCON_APPLID	Server region CICS APPLID - for diagnostic purposes only
(18)	CHARACTER	8	DTCON_FILE_NAME	File name in server region - for diagnostic purposes only

---

DTLPS Data tables connection anchor blocks

DTHDR_BLOCK, the Data Tables Header Block, is a unique CICS lifetime block which is getmained by CICS data tables initialization and referenced by CICS data tables loading and record access services. It contains heads of chains and other information which occurs once per CICS region, plus a storage area which is used by the record retrieval module DFHDTRE for its working storage.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	DTHDR_BLOCK	Header Block
(0)	CHARACTER	16	DTHDR_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTHDR_LEN	Length of header block
(2)	CHARACTER	1	DTHDR_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTHDR_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTHDR_ID	Eye catcher 'HEADER'
(10)	ADDRESS	4	DTHDR_DTFOR_EP	DFHDTFOR module entry point for diagnostic purposes
(14)	ADDRESS	4	DTHDR_RECMAN_EP	Record manager entry point (DFHDTRM, loaded to address)
(18)	CHARACTER	16	DTHDR_TABLE_INFO	Table block information
(18)	ADDRESS	4	DTHDR_TABLE_ HEAD	Head of active table chain
(1C)	ADDRESS	4	DTHDR_TABLE_ POOL	Table block cell pool id
(20)	ADDRESS	4	DTHDR_TABLE_ FREE	Head of free chain
(24)	FULLWORD	4	DTHDR_TABLE_ COUNT	Number of blocks in use
(28)	CHARACTER	20	DTHDR_FILE_INFO	File block information
(28)	ADDRESS	4	DTHDR_FILE_HEAD	Head of active file chain
(2C)	ADDRESS	4	DTHDR_FILE_POOL	File block cell pool id
(30)	ADDRESS	4	DTHDR_FILE_FREE	Head of free chain
(34)	FULLWORD	4	DTHDR_FILE_ COUNT	Number of blocks in use
(38)	FULLWORD	4	DTHDR_MAX_ ATTRS_LEN	File attribute suffix size
(3C)	FULLWORD	4	DTHDR_LOAD_ID	Unique identifier which is allocated to each table load task, always contains the most recently allocated id
(40)	ADDRESS	4	DTHDR_BACKOUT_ POOL	Backout cell pool id
(44)	UNSIGNED	4	DTHDR_PRIMARY_ ALET	ALET used to access table index and entry data areas in the server address space, changed when any table is closed to interrupt active requests so that retry can revalidate the connections
(48)	ADDRESS	4	DTHDR_DATA_ SPACE_PTR	Address of data space block
(4C)	ADDRESS	4	*	Reserved, alignment to dword
(50)	CHARACTER	*	DTHDR_RE_WORK	DFHDTRE working storage

At Data Tables FOR initialization, DFHDTINS getmains and initializes DTDUM_BLOCK. This block represents a dummy table and must always overlay the first part of DTTBL_BLOCK so that the pointer to the header block is at the same offset in both control blocks. Its address is passed in DTP_TABLE_TOKEN whenever DFHDTUP is called for a commit/backout request, and it allows commit and backout to find the header block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	36	DTDUM_BLOCK	Dummy recovery blk
(0)	CHARACTER	24	DTDUM_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTDUM_LEN	Length of table block
(2)	CHARACTER	1	DTDUM_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTDUM_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTDUM_ID	Eye catcher 'DUMMY'
(10)	CHARACTER	8	DTDUM_NAME	Unused, matches table block
(18)	CHARACTER	8	DTDUM_CHAIN	Unused, matches table block
(18)	ADDRESS	4	DTDUM_NEXT	Unused, matches table block

Offset Hex	Type	Len	Name (Dim)	Description
(1C)	UNSIGNED	4	DTDUM_CHANGES	Unused, matches table block
(20)	ADDRESS	4	DTDUM_HEADER_PTR	Pointer back to header block

DTTBL_BLOCK, the DT Table Block, is the control block which describes a table and its associated index and record storage. The first few fields should never be moved without also changing DTDUM_BLOCK, because the pointer to the header block must remain at the same offset in both.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	232	DTTBL_BLOCK	Data Tables Table Block
(0)	CHARACTER	24	DTTBL_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTTBL_LEN	Length of table block
(2)	CHARACTER	1	DTTBL_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTTBL_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTTBL_ID	Eye catcher 'TABLE'
(10)	CHARACTER	8	DTTBL_NAME	Name of file which initiated the creation of the table
(18)	CHARACTER	8	DTTBL_CHAIN	Align for block-concurrent fetch so change count can be used to validate chain field
(18)	ADDRESS	4	DTTBL_NEXT	Next in active or free chain or zero at end of chain
(1C)	UNSIGNED	4	DTTBL_CHANGES	Counter updated whenever a change is made to the table state or table contents, and also when the block is freed
(20)	ADDRESS	4	DTTBL_HEADER_PTR	Pointer back to header block
(24)	BITSTRING	1	DTTBL_FLAGS	Table type and state flags
	1...		DTTBL_CMT	On if CICS maintained table, Off if user maintained (UMT)
	.1..		DTTBL_RECOVERABLE	Table is a recoverable UMT
	..1.		DTTBL_INCOMPLETE	One or more gaps in table (CMT only)
	...1		DTTBL_LOAD_EOF	Set by the END_LOAD service when loader has reached EOF
	.... 1...		DTTBL_LOAD_GAP	The previous record was discarded during loading, so the next accepted record will need a gap before it
	.... .1..		DTTBL_LOAD_DISC	A record with a key above the highest loaded key was discarded since the previous loading request, so a gap is needed if the next loaded record has a higher key
	.... ..1.		DTTBL_ADD_GAP	Within add processing, this indicates whether the entry is being added within a gap
	1		*	Reserved
(25)	BITSTRING	1	DTTBL_T_FLAGS	Table shared access flags
	1...		DTTBL_AVAILABLE	Table available for access. Set when table reaches a stage at which it is available for shared access (for a CMT - when the load load has been initiated, for a UMT - at completion of loading). Never turned off again until table is closed.
	.111 1111		*	Reserved
(26)	CHARACTER	2	*	Reserved for alignment
(28)	FULLWORD	4	DTTBL_FILE_COUNT	Number of associated files
(2C)	ADDRESS	4	DTTBL_DSNAME_PTR	Source data set name pointer
(30)	FULLWORD	4	DTTBL_DSNAME_LEN	Length of data set name
(34)	FULLWORD	4	DTTBL_LOAD_ID	Identifying counter of the valid loading task for this table
(38)	CHARACTER	16	DTTBL_STATS	External statistics about internal (loading) requests
(38)	UNSIGNED	4	DTTBL_LOAD_ COUNT	Requests to load a record
(3C)	UNSIGNED	4	DTTBL_REJECT_ COUNT	Loads rejected by user exit
(40)	UNSIGNED	4	DTTBL_FULL_ COUNT	Loads failed due to full tbl
(44)	UNSIGNED	4	DTTBL_ENTRY_HWM	Entry count high water mark
(48)	FULLWORD	4	DTTBL_KEY_LEN	Length of record key
(4C)	FULLWORD	4	DTTBL_KEY_OFFSET	Offset of key within record
(50)	FULLWORD	4	DTTBL_MAX_RECLEN	Maximum record length
(54)	ADDRESS	4	DTTBL_LOAD_ HIGH_KEY	Address of copy of highest key accepted during loading, which must be changed only by switching the pointer to a new copy, to allow for concurrent read access
(58)	ADDRESS	4	DTTBL_LOAD_ DISC_KEY	Address of copy of lowest discarded key above previous highest loaded key (valid if discarded key flag is set), also used as alternate area for highest loaded key area, swapped over at each change
(5C)	CHARACTER	16	DTTBL_ENTRY_INFO	Entry information, primarily for record manager DFHDTRM
(5C)	ADDRESS	4	DTTBL_ENTRY_ALET_PTR	Table entry ALET pointer
(60)	ADDRESS	4	DTTBL_ENTRY_ POOL	Record entry pool token
(64)	FULLWORD	4	DTTBL_ENTRY_ COUNT	Number of entries in use
(68)	FULLWORD	4	DTTBL_ENTRY_ LIMIT	Limit specified for table
(6C)	CHARACTER	12	DTTBL_ADD_SAVE	Temporary saved position within add processing while locating the previous record
(6C)	UNSIGNED	4	* (3)	Position needs 3 fullwords
(78)	CHARACTER	20	DTTBL_INDEX_INFO	Index information, primarily for index manager DFHDTIX
(78)	ADDRESS	4	DTTBL_INDEX_ ROOT	Root node for index tree
(7C)	ADDRESS	4	DTTBL_INDEX_ ALET_PTR	Index storage ALET pointer
(80)	ADDRESS	4	DTTBL_INDEX_ POOL	Index cell pool token
(84)	FULLWORD	4	DTTBL_INDEX_ COUNT	Index cells in use
(88)	FULLWORD	4	DTTBL_INDEX_HWM	High water index cells
(8C)	CHARACTER	44	DTTBL_DATA_INFO	Data storage and data space information, primarily for DFHDTDM and DFHDTDA
(8C)	FULLWORD	4	DTTBL_DATA_ SPACE	Index within DTDSP_VECTOR of entry for the data space to which this table is assigned
(90)	ADDRESS	4	DTTBL_DATA_ ALET_PTR	Data space ALET pointer
(94)	ADDRESS	4	DTTBL_DATA_HEAD	Head of data frame chain

Offset Hex	Type	Len	Name (Dim)	Description
(98)	FULLWORD	4	DTTBL_DATA_ FRAME	Size of each frame
(9C)	ADDRESS	4	DTTBL_DATA_ START	Origin of first frame area
(A0)	ADDRESS	4	DTTBL_DATA_NEXT	Next unallocated frame
(A4)	ADDRESS	4	DTTBL_DATA_END	End of current frame area
(A8)	FULLWORD	4	DTTBL_DATA_SIZE	Total data storage in use
(AC)	ADDRESS	4	DTTBL_DATA_FREE	Head of free frame chain
(B0)	FULLWORD	4	DTTBL_DATA_ COUNT	Number of data areas in use
(B4)	FULLWORD	4	DTTBL_DATA_HWM	High water data area count
(B8)	FULLWORD	4	DTTBL_RETRY_ COUNT	Shared read retry count

The next field should always be addressed indirectly using DTTBL_DSNAME_PTR except when it is first set up. This allows new fields to be added in front of it, and means that it can be removed if it becomes unnecessary to store the DSN in the table.

Offset Hex	Type	Len	Name (Dim)	Description
(BC)	CHARACTER	44	DTTBL_DSNAME	Source data set name

DTFIL_BLOCK is a data tables file block. There is one such block for every UMT, and one for each file resource that refers to a source data set where one of the files is defined as a CMT.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	DTFIL_BLOCK	Data Tables File Block
(0)	CHARACTER	24	DTFIL_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTFIL_LEN	Length including attributes
(2)	CHARACTER	1	DTFIL_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTFIL_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTFIL_ID	Eye catcher 'FILE'
(10)	CHARACTER	8	DTFIL_NAME	File resource name
(18)	CHARACTER	8	DTFIL_CHAIN	Align for block-concurrent fetch so reuse count can be used to validate chain field
(18)	ADDRESS	4	DTFIL_NEXT	Next in active or free chain or zero at end of chain
(1C)	UNSIGNED	4	DTFIL_REUSE_ COUNT	Allocate and release count - odd when file block is in the active file chain (i.e. DTFIL_NEXT is valid for an active chain scan)
(20)	ADDRESS	4	DTFIL_TABLE_PTR	Pointer to table block
(24)	BITSTRING	1	DTFIL_FLAGS	File-related status flags

-- Shared access to a file uses the DTFIL_ENABLED and DTFIL_CONTINUE flags. DTFIL_ENABLED flag on means file enabled for new requests. This flag is tested on shared access when the request specifies TEST_ENABLE, but is ignored otherwise. The feature should never set this flag to disabled unless it knows from file control that the file really is disabled. The flag is therefore set ON when the file is opened, and reset to ON or OFF on a SET_ENABLEMENT call. If the flag is OFF then new requests will fail with a DISABLED exception. DTFIL_CONTINUE flag on means old requests can continue. When this flag is ON, existing requests to the file can continue. If the flag is OFF then all requests will fail with a DISABLED exception, regardless of whether or not they are continuations of existing requests (which do not test DTFIL_ENABLED). This flag will always be ON unless a FORCE DISABLE is issued, when it will be set to OFF. A subsequent ENABLE request will turn it back on. The flag is set ON when the file block is opened. ------

Offset Hex	Type	Len	Name (Dim)	Description
	1...		DTFIL_ENABLED	Enabled for new requests
	.1..		DTFIL_INITIATOR	File initiated the table
	..1.		DTFIL_CONTINUE	Old requests can continue
	...1 1111		*	Reserved
(25)	BITSTRING	1	DTFIL_A_FLAGS	File shared access flags
	1...		DTFIL_AVAILABLE	Available for shared access. When set, file is visible. Set on once the enablement state of the file is known, never turned off until the file is closed.
	.111 1111		*	Reserved
(26)	CHARACTER	2	*	Reserved for alignment
(28)	FULLWORD	4	DTFIL_ATTRS_LEN	Length of attributes package
(2C)	CHARACTER	*	DTFIL_ATTRS	Saved file attributes

DTRPS Data tables remote sharing anchor block

DTRHD_BLOCK, the Data Tables Remote Header Block, is a unique CICS lifetime block which is getmained by CICS data tables remote initialization. It contains information which occurs once per application region which has connections to shared data tables in other regions. In the current implementation, this only consists of a pointer used for diagnostic purposes.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	DTRHD_BLOCK	Remote Header Block
(0)	CHARACTER	16	DTRHD_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTRHD_LEN	Length of remote header
(2)	CHARACTER	1	DTRHD_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTRHD_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTRHD_ID	Eye catcher 'REMHEAD'
(10)	ADDRESS	4	*	Reserved for future use
(14)	ADDRESS	4	DTRHD_DTAOR_EP	DFHDTAOR module entry point for diagnostic purposes

DTSPS Data tables SVC routine anchor blocks

DTSYS_ANCHOR, the Data Tables System Anchor, is allocated once within an MVS image. It primarily provides an anchor point to enable code running in one address space to find out about data table servers running in other address spaces.
Each region using data tables initially accesses the system anchor via the internal CICS QSSCT chain starting at SSCTSUS2 in the "CICS" SSCVT, then saves the address in the region anchor for subsequent use. The address also appears in the server element for use by the EOM RESMGR routine.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	DTSYS_ANCHOR	Data Tables System Anchor
(0)	CHARACTER	16	DTSYS_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTSYS_LEN	Length of system anchor
(2)	CHARACTER	1	DTSYS_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTSYS_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTSYS_ID	Eye catcher 'SYSTEM'
(10)	CHARACTER	8	DTSYS_ACTIVE_ CLOCK	STCK value updated when files become available for shared access
(18)	ADDRESS	4	DTSYS_SERVER_ HEAD	Head of active server chain
(1C)	UNSIGNED	4	DTSYS_CONNECTS_ IN_FLIGHT	Number of in-flight CONNECT requests in this MVS image that cannot tolerate termination of their server

DTRGN_ANCHOR, the Data Tables Region Anchor, is allocated once per region which is using shared data tables support, and is located via AFDTRGNP for the appropriate CICS QR TCB. It provides a common anchor for the data areas used by supervisor code for data tables server and connection processing. Note that the offset of DTRGN_LOOKUP_EP is relied on by code outside the SVC routine, and must remain fixed for any new version.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	76	DTRGN_ANCHOR	Data Tables Region Anchor
(0)	CHARACTER	16	DTRGN_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTRGN_LEN	Length of region anchor
(2)	CHARACTER	1	DTRGN_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTRGN_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTRGN_ID	Eye catcher 'REGION'
(10)	ADDRESS	4	DTRGN_SYSTEM_PTR	Address of system anchor
(14)	CHARACTER	12	DTRGN_CONNECT_ INFO	Connected region information
(14)	ADDRESS	4	DTRGN_REMOTE_ PTR	Remote header block address set from global token passed on remote initialization
(18)	ADDRESS	4	DTRGN_LOOKUP_EP	Connect vector look-up entry point (DFHDTCV in ECSA) - CAUTION - THIS OFFSET MUST NOT CHANGE - see preceding block comment.

Offset Hex	Type	Len	Name (Dim)	Description
(1C)	ADDRESS	4	DTRGN_CONNECT_ PTR	Connect block address, set up at remote initialization
(20)	CHARACTER	44	DTRGN_SERVER_ INFO	Server region information
(20)	ADDRESS	4	DTRGN_HEADER_ PTR	Local header block address, set from global token passed on local initialization
(24)	ADDRESS	4	DTRGN_RECMAN_EP	Record manager entry point, loaded during server initialization
(28)	ADDRESS	4	DTRGN_SERVER_ PTR	Server element address, set during server logon
(2C)	UNSIGNED	4	DTRGN_EOM_TOKEN	EOM RESMGR token
(30)	CHARACTER	8	DTRGN_HOME_ STOKEN	Home address space STOKEN
(38)	ADDRESS	4	DTRGN_ALET_ LIST_PTR	Start of first section of list of PASN ALETs added by DTSVC, for DELETE validation
(3C)	ADDRESS	4	DTRGN_EXIT_ WORKA_PTR	Address of work area for SYNCH exit to issue trial ALESERV for STOKEN checks
(40)	BITSTRING	1	DTRGN_FLAGS	Flag byte
	1...		DTRGN_TRANSWAP	SYSEVENT TRANSWAP was done
	.1..		DTRGN_EOM_ RESMGR_ DELETE_ACTIVE	EOM RESMGR DELETE might be in progress
	..11 1111		*	Reserved
(41)	CHARACTER	3	*	Reserved for alignment
(44)	FULLWORD	4	DTRGN_DTAM_ LENGTH	Length of DFHDTAM, set if CICS has loaded DTAM, zero if it is in the LPA
(48)	ADDRESS	4	DTRGN_DTAM_ ORIGIN	Origin of DFHDTAM in storage, set if CICS has loaded DTAM, zero if it is in the LPA

DTSRV_ELEMENT, a Data Tables Server element, is created in ECSA when a server region logs on. Its address is stored in the region anchor, and when it is active it can be located from other address spaces via a chain from the the system anchor. It contains the information needed to connect to an active server from another address space.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	DTSRV_ELEMENT	Data Tables Server Element
(0)	CHARACTER	24	DTSRV_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTSRV_LEN	Length of block
(2)	CHARACTER	1	DTSRV_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTSRV_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTSRV_ID	Eye catcher 'SERVER'
(10)	CHARACTER	8	DTSRV_APPLID	Server generic CICS APPLID
(18)	ADDRESS	4	DTSRV_NEXT	Chain to next, zero if last
(1C)	ADDRESS	4	DTSRV_SYSTEM_PTR	Address of system anchor - Zero if this server element is neither in the active chain nor being used by any in-flight CONNECT requests
(20)	UNSIGNED	2	DTSRV_ASID	Server address space id
(22)	UNSIGNED	2	DTSRV_LX	Server PC linkage index - 1st bit is 1 if this server does not currently own an LX
(24)	UNSIGNED	4	DTSRV_ET_TOKEN	Server PC entry table token
(28)	ADDRESS	4	DTSRV_SEC_EP	Connect security entry point
(2C)	ADDRESS	4	DTSRV_SEC_TOKEN	Connect security block token - Zero if this server is not enforcing file security
(30)	FULLWORD	4	DTSRV_DTAM_ LENGTH	Length of DFHDTAM, set if CICS has loaded DTAM, zero if it is in the LPA
(34)	ADDRESS	4	DTSRV_DTAM_ ORIGIN	Origin of DFHDTAM in storage, set if CICS has loaded DTAM, zero if it is in the LPA

DTXPS Data tables security anchor block

DTSEC_BLOCK, the Data Tables Security Block, is allocated in ECSA by connect security initialization, called during server logon processing. It contains information from the server address space which will be needed for security checks at connect time, when the server private region is not accessible. It is pointed to by the security token in the server element.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	DTSEC_BLOCK	Data Tables Security Block
(0)	CHARACTER	16	DTSEC_PREFIX	Standard CICS prefix
(0)	HALFWORD	2	DTSEC_LEN	Length of security block
(2)	CHARACTER	1	DTSEC_ARROW	Eye catcher '>'
(3)	CHARACTER	5	DTSEC_DFHDT	Eye catcher 'DFHDT'
(8)	CHARACTER	8	DTSEC_ID	Eye catcher 'SECURITY'
(10)	CHARACTER	8	DTSEC_SERVER_ USERID	Security userid for server region, binary zero if none
(18)	CHARACTER	8	DTSEC_DEFAULT_ USERID	Server region default userid
(20)	CHARACTER	9	DTSEC_RESNAME_ PREFIX	Resource name prefix including final '.'
(29)	UNSIGNED	1	DTSEC_RESNAME_ PREFIX_LENGTH	Length of resource name prefix, zero if none
(2A)	UNSIGNED	1	*	Reserved
(2B)	UNSIGNED	1	DTSEC_FC_ CLASS_NAME_LENGTH	Length of security class name for server's files
(2C)	CHARACTER	8	DTSEC_FC_ CLASS_NAME	Security class name for server's files

DUFC Dump formatting communication area

DFHDUFC - dump formatting - communication area etc.
Dump formatting communication area.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	108	DUF_COM	
(0)	ADDRESS	4	DUF_PRDMP_ PARMLIST_PTR	
(4)	ADDRESS	4	DUF_AFCB_PTR	
(8)	ADDRESS	4	*	
(C)	ADDRESS	4	*	
(10)	ADDRESS	4	DUF_DOMAIN_ TABLE_PTR	
(14)	BITSTRING	1	*	
	1...		DUF_UPPERCASE_ REQ	
	.111 1111		*	
(15)	CHARACTER	3	*	
(18)	FULLWORD	4	*	unused
(1C)	FULLWORD	4	*	unused
(20)	FULLWORD	4	*	unused
(24)	FULLWORD	4	*	unused
(28)	CHARACTER	48	DUF_NDX_HEAD	
(58)	ADDRESS	4	DUF_NDX_FREEHEAD	
(5C)	ADDRESS	4	DUF_ERB_IHEAD	
(60)	ADDRESS	4	DUF_ERB_IFREE	
(64)	ADDRESS	4	DUF_ERB_EHEAD	
(68)	ADDRESS	4	DUF_ERB_EFREE	
(6C)	CHARACTER		*	

Domain table.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	160	DUF_DOMAIN_TABLE	
(0)	ADDRESS	4	DUF_DOMAIN_ ANCHOR (40)	
(A0)	CHARACTER		*	

Control block index entry.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	48	NDX	
(0)	ADDRESS	4	NDX_NEXT	-> next in address order
(4)	ADDRESS	4	NDX_NEXT2	-> next in name order
(8)	ADDRESS	4	NDX_BLOCK_ ADDRESS	
(C)	FULLWORD	4	NDX_BLOCK_LENGTH	
(10)	FULLWORD	4	NDX_PAGE_NUMBER	
(14)	CHARACTER	25	NDX_BLOCK_NAME	name.resource
(2D)	CHARACTER	3	*	reserved
(30)	CHARACTER		*	

TMP Browse Block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	TBB	
(0)	CHARACTER	4	TBB_EYECATCHER	
(4)	ADDRESS	4	TBB_DIR_ ELEMENT_ADDRESS	

Error index block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	264	ERB	
(0)	ADDRESS	4	ERB_NEXT	-> next error block
(4)	FULLWORD	4	ERB_INDEX	number in this block
(8)	FULLWORD	4	ERB_PAGE_NUMBER (64)	page number array

Constants

Len	Type	Value	Name	Description
4	CHARACTER	>TBB	TBB_EYECATCHER_ VALUE	

DUFP Parameter area declarations

DFHDUFP - dump formatting routines - parameter declarations.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	280	DUF_PARMS	
(0)	CHARACTER	16	DUF_PREFIX	
(0)	HALFWORD	2	DUF_LENGTH	
(2)	CHARACTER	1	DUF_ARROW	
(3)	CHARACTER	3	DUF_DFH	
(6)	CHARACTER	2	DUF_DOMID	
(8)	CHARACTER	8	DUF_BLCK_NAME	
(10)	ADDRESS	4	DUF_COM_PTR	
(14)	CHARACTER	200	DUF_USER_PARMS	
(14)	UNSIGNED	1	DUF_FUNCTION	
(15)	BITSTRING	1	DUF_FLAGS	
	1...		DUF_EJECT	
	.1..		DUF_SPACE_ BEFORE	
	..1.		DUF_SPACE_ AFTER	
	...1		DUF_ALLOW_ZERO	
	.... 1...		DUF_LONG_ NAME_X	
	.... .1..		*	
	.... ..1.		DUF_BLOCK_RESOURCE2_X	
	1		*	
(16)	UNSIGNED	1	DUF_INDEX_ENTRY_TYPE	
(16)	UNSIGNED	1	DUF_TMP_TABLE	
(17)	UNSIGNED	1	DUF_SEVERITY_ LEVEL	
(18)	UNSIGNED	1	DUF_MESSAGE_ TYPE	
(19)	UNSIGNED	1	DUF_BOUNDARY	
(1A)	UNSIGNED	1	*	
(1B)	UNSIGNED	1	*	
(1C)	FULLWORD	4	DUF_RC	
(20)	ADDRESS	4	DUF_BLOCK_ ADDRESS	
(24)	FULLWORD	4	DUF_BLOCK_ LENGTH	
(28)	ADDRESS	4	DUF_SET_PTR	
(2C)	ADDRESS	4	DUF_ANCHOR_PTR	
(30)	ADDRESS	4	DUF_LIST_TOKEN	
(30)	ADDRESS	4	DUF_BROWSE_ TOKEN	
(34)	ADDRESS	4	DUF_ADDRESS	
(34)	ADDRESS	4	DUF_TABLE_ENTRY_ADDRESS	
(38)	FULLWORD	4	DUF_OFFSET	
(3C)	CHARACTER	24	DUF_LONG_NAME	
(3C)	CHARACTER	8	DUF_BLOCK_NAME	
(44)	CHARACTER	16	DUF_BLOCK_RESOURCE	
(54)	FULLWORD	4	DUF_BLOCK_TITLE_LENGTH	
(54)	FULLWORD	4	DUF_INDEX_ENTRY_TEXT_LENGTH	
(54)	FULLWORD	4	DUF_MESSAGE_TEXT_LENGTH	
				@BA22329A
(58)	CHARACTER	132	DUF_LINE	
(58)	CHARACTER	112	DUF_BLOCK_ TITLE	
(58)	CHARACTER	40	DUF_INDEX_ENTRY_TEXT	
(58)	CHARACTER	30	DUF_MESSAGE_ TEXT	
(C8)	CHARACTER	8	DUF_BLOCK_RESOURCE2	
(DC)	CHARACTER		*	

Offset Hex	Type	Len	Name (Dim)	Description
(DC)	BITSTRING	4	DUF_FORMAT_LEVEL	
	1...		DUF_FORMAT_ SUMMARY	
	.1..		DUF_FORMAT_ BLOCKS	
	..1.		DUF_FORMAT_ CHECKING	
(DC)	BITSTRING	3	*	
(E0)	CHARACTER	33	DUF_TIME_DATE	
(E0)	CHARACTER	17	DUF_TIME_DATE_FORMAT	
(F1)	CHARACTER	8	DUF_TIME_ DATE_STCK	
(F9)	CHARACTER	8	DUF_DUMP_HEADER_STCK	
(101)	CHARACTER	3	*	
(104)	ADDRESS	4	DUF_TRFCA_PTR	
(108)	UNSIGNED	2	DUF_LINES_LEFT_ON_PAGE	
				*
(10A)	CHARACTER	1	*	
(10A)	BITSTRING	1	DUF_FLAGS2	
	1...		DUF_PF3_ PRESSED	
(10B)	CHARACTER	1	*	
(10C)	CHARACTER	8	DUF_READ_TOKEN	
(10C)	ADDRESS	4	DUF_READ_PTR	
(110)	FULLWORD	4	DUF_READ_INDEX	
(114)	ADDRESS	4	DUF_DUFF_PTR	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	1	DUF_FORMAT_BLOCK	
4	DECIMAL	2	DUF_GET_BLOCK	
4	DECIMAL	3	DUF_PRINT_LINE	
4	DECIMAL	4	DUF_PRINT_MESSAGE	
4	DECIMAL	5	DUF_CREATE_LIST	
4	DECIMAL	6	DUF_DELETE_LIST	
4	DECIMAL	7	DUF_ADD_LIST	
4	DECIMAL	8	DUF_ADD_INDEX_ ENTRY	
4	DECIMAL	9	DUF_TMP_START_ BROWSE	
4	DECIMAL	10	DUF_TMP_GET_NEXT	
4	DECIMAL	11	DUF_TMP_END_BROWSE	
4	DECIMAL	12	DUF_FORMAT_MAIN_STORAGE	
4	DECIMAL	13	DUF_FORMAT_STCK	
4	DECIMAL	14	DUF_START_ READ_LIST	
4	DECIMAL	15	DUF_READ_LIST	
4	DECIMAL	16	DUF_ADD_LIST_ REVERSE	
4	DECIMAL	17	DUF_READ_ LIST_REVERSE	
4	DECIMAL	18	DUF_START_READ_LIST_REVERSE	
4	DECIMAL	19	DUF_CREATE_LIST_REVERSE	
Index entry types.				
4	DECIMAL	1	DUF_INDEX_ENTRY_TYPE_KEYWORD	
4	DECIMAL	2	DUF_INDEX_ENTRY_TYPE_BLOCK	
4	DECIMAL	3	DUF_INDEX_ENTRY_TYPE_TEXT	
Message types.				
4	DECIMAL	1	DUF_MSG_ZERO_ POINTER	
4	DECIMAL	2	DUF_MSG_INVALID_POINTER	
4	DECIMAL	3	DUF_MSG_ZERO_ ADDRESS	
4	DECIMAL	4	DUF_MSG_INVALID_ADDRESS	
4	DECIMAL	5	DUF_MSG_LOOP_DETECTED	
4	DECIMAL	6	DUF_MSG_FORMATTING_ERROR	
4	DECIMAL	7	DUF_MSG_INVALID_EYECATCHER	
4	DECIMAL	8	DUF_MSG_TMP_START_BROWSE	
4	DECIMAL	9	DUF_MSG_TMP_ GET_NEXT	
4	DECIMAL	10	DUF_MSG_UNREFERENCED_PAGE	
4	DECIMAL	11	DUF_MSG_INVALID_DATA_LEN	
4	DECIMAL	12	DUF_MSG_SAA1_ INVALID	
4	DECIMAL	13	DUF_MSG_SAA2_ INVALID	
4	DECIMAL	14	DUF_MSG_SAAS_ INVALID	
4	DECIMAL	15	DUF_MSG_SAAS_ DIFFER	

Len	Type	Value	Name	Description
4	DECIMAL	16	DUF_MSG_INVALID_ DATA	@BA22329A
Message severity level values.				
4	DECIMAL	1	DUF_SEVERITY_ LEVEL_I	
4	DECIMAL	2	DUF_SEVERITY_ LEVEL_E	
TMP table types.				
4	DECIMAL	4	DUF_TMP_TABLE_PFT	
4	DECIMAL	5	DUF_TMP_TABLE_FCT	
4	DECIMAL	6	DUF_TMP_TABLE_DCT	
4	DECIMAL	7	DUF_TMP_TABLE_TCTE	
4	DECIMAL	8	DUF_TMP_TABLE_TCTN	
4	DECIMAL	9	DUF_TMP_TABLE_TCTS	
4	DECIMAL	10	DUF_TMP_TABLE_AFCT	
4	DECIMAL	11	DUF_TMP_TABLE_DSN	
4	DECIMAL	12	DUF_TMP_TABLE_DSNA	
4	DECIMAL	13	DUF_TMP_TABLE_PRT	
4	DECIMAL	15	DUF_TMP_TABLE_TCNT	
4	DECIMAL	15	DUF_TMP_TABLE_DUMY	
4	DECIMAL	16	DUF_TMP_TABLE_AITM	
Return codes				
4	DECIMAL	0	DUF_OK	
4	DECIMAL	1	DUF_INVALID_ ADDRESS	
4	DECIMAL	2	DUF_NOT_FOUND	
4	DECIMAL	3	DUF_FORMATTING_ ERROR	
4	DECIMAL	4	DUF_DUPLICATE_ ADDRESS	
4	DECIMAL	5	DUF_END_BROWSE	
4	DECIMAL	6	DUF_TMP_START_ BROWSE_ERROR	
4	DECIMAL	7	DUF_TMP_GET_ NEXT_ERROR	
4	DECIMAL	8	DUF_INVALID_ BROWSE_TOKEN	
4	DECIMAL	9	DUF_INVALID_ DATA_LEN	
4	DECIMAL	10	DUF_QUIT_JOB	

D2CSB Csub block

```
CONTROL BLOCK NAME = DFHD2CSB
DESCRIPTIVE NAME = CICS DB2 Subtask block
FUNCTION =
    The DFHD2CSB block contains state data for the CICS-DB2
    subtask. It is also used as working storage by the subtask.
LIFETIME =
    A DFHD2CSB is getmained when a Subtask TCB is attached.
    It is freemained when a subtask is detached.
LOCATION =
    DFHD2CSB blocks are chained together off the DFHD2GLB and
    off either a DB2ENTRY or the pool or command thread section
    of the DFHD2GLB. There are a number of chains. Which chain
    a DFHD2CSB is on is governed by the state of the Thread.
    There are chains for free Tcbs, free protected threads, and
    active threads.
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS = none
 MODULE TYPE = Control block definition
  DFHD2CSB block
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	740	DFHD2CSB	
(0)	CHARACTER	16	CSB_PREFIX	standard Prefix
(0)	HALFWORD	2	CSB_LENGTH	
(2)	CHARACTER	14	CSB_EYE	>DFHD2CSB
(10)	CHARACTER	8	CSB_CLOCK	STCK for unique name
(18)	ADDRESS	4	CSB_GLB_ADDRESS	global block address
(1C)	ADDRESS	4	CSB_RCT_ADDRESS	RCT entry block address
(20)	ADDRESS	4	CSB_LOT_ADDRESS	Life of task block addr
(24)	ADDRESS	4	CSB_TCB_ADDRESS	subtask TCB
(28)	UNSIGNED	4	CSB_ECB	subtask ECB
(2C)	UNSIGNED	4	CSB_TERMINATE_ ECB	terminate ECB for EX2
Active thread chain				
(30)	ADDRESS	4	CSB_ACTIVE_PREV	prev CSUB on active chain
(34)	ADDRESS	4	CSB_ACTIVE_NEXT	next CSUB on active chain

Offset Hex	Type	Len	Name (Dim)	Description
Free protected thread chain anchored off RCTE				
(38)	ADDRESS	4	CSB_RCT_ PTHREAD_PREV	prev CSUB on free protect
(3C)	ADDRESS	4	CSB_RCT_ PTHREAD_NEXT	next CSUB on free protect
Free protected thread chain anchored off D2GLB				
(40)	ADDRESS	4	CSB_GLB_ PTHREAD_PREV	prev CSUB on Global fprot
(44)	ADDRESS	4	CSB_GLB_ PTHREAD_NEXT	next CSUB on Global fprot
Free TCB chain anchored off RCTE				
(48)	ADDRESS	4	CSB_RCT_TCB_PREV	prev CSUB on free tcb
(4C)	ADDRESS	4	CSB_RCT_TCB_NEXT	next CSUB on free tcb
Global Free TCB chain anchored of D2GLB				
(50)	ADDRESS	4	CSB_GLB_TCB_PREV	prev CSUB on Global free
(54)	ADDRESS	4	CSB_GLB_TCB_NEXT	next CSUB on Global free
Attach/Detach chain (singly linked)				
(58)	ADDRESS	4	CSB_ATTACH_ DETACH_NEXT	Next CSUB on chain
(5C)	CHARACTER	8	CSB_PLAN_NAME	plan name
(64)	CHARACTER	8	CSB_PRIMARY_ AUTH_NAME	auth name to sign on
(6C)	CHARACTER	8	CSB_SECONDARY_ AUTH_NAME	secondary auth to sign on
(74)	CHARACTER	12	CSB_CORRELATION_ ID	CSUB Correlation id
(74)	CHARACTER	4	CSB_TYPE	type ENTR/POOL/COMD
(78)	CHARACTER	4	CSB_TRANSID	transaction id
(7C)	CHARACTER	4	CSB_THREAD_ NUMBER_DEC	thread number in decimal
(80)	ADDRESS	4	CSB_ACEE_ADDRESS	address of ACEE
(84)	CHARACTER	8	CSB_SIGNON_TIME	STCK at time of signon
(8C)	CHARACTER	22	CSB_ACCOUNT_ TOKEN	accounting corr.token
(8C)	CHARACTER	8	CSB_ACCOUNT_ NETNAME	netname
(94)	CHARACTER	8	CSB_ACCOUNT_ LUNAME	luname
(9C)	CHARACTER	6	CSB_ACCOUNT_ CLOCK	middle of STCK
(A2)	BITSTRING	1	CSB_ACCOUNT_ TOKEN_FLAG	accounting corr.flag
	1...		CSB_ACCOUNT_ TOKEN_ACTIVE	accounting corr.active
	.111 1111		*	reserved
(A3)	BITSTRING	1	CSB_CTL1	connection control flag
	1...		CSB_ATTACH_TASK	attach subtask
	.1..		CSB_DETACH_TASK	detach task
	..1.		CSB_TASK_ ATTACHED_OK	attach ok
	...1		CSB_TERMINATE_ TASK	terminate subtask
	.... 1...		CSB_TASK_ TERMED_OK	subtask terminated OK
	.... .1..		CSB_TASK_ TERMED_ABNORMAL	subtask abnormal end
	11		*	
(A4)	BITSTRING	1	CSB_CTL2	connection control flag
	1...		CSB_PROTECTED_ THREAD	protected thread
	.1..		CSB_INITIAL_ STATE	initial state thread ind.
	..1.		CSB_CURSOR	cursor hold on
	...1		CSB_AVAIL_ ASSIGN	available for reuse
	.... 1...		CSB_TERM_THREAD	terminate thread
	.... .1..		CSB_THREAD_ CREATED	thread created
	.... ..1.		CSB_SUBTASK_ IN_DB2	subtask is in DB2
	1		CSB_SUBTASK_ RUNNING	subtask is running
(A5)	CHARACTER	1	CSB_CHAP	CICS task priority
(A6)	UNSIGNED	2	CSB_THREAD_ NUMBER	Binary form of thread num

Offset Hex	Type	Len	Name (Dim)	Description
(A8)	CHARACTER	8	CSB_PRIMARY_AUTH_SAVEAREA	auth savarea
(B0)	CHARACTER	8	CSB_SECONDARY_AUTH_SAVEAREA	secondary auth savearea
(B8)	CHARACTER	16	CSB_NETWORK_ID	blank network id
(C8)	ADDRESS	4	CSB_WLM_ PERF_TOKEN	CICS WLM perf blk token
(CC)	CHARACTER	48	CSB_FRB	FRB area
(FC)	CHARACTER	72	CSB_SAVEAREA	subtask save area
(144)	CHARACTER	80	CSB_WORKAREA	work area
(194)	CHARACTER	52	CSB_ERROR_BUFFER	error resource buffer
(1C8)	UNSIGNED	4	CSB_REQUEST_ NUMBER	request num HWM for trace
(1CC)	ADDRESS	4	CSB_CURRENT_TRACE_ENTRY	Pointer to trace entry
Trace table for subtask				
(1D0)	CHARACTER	16	CSB_TRACE_HEAD	start of trace eyecatcher
(1E0)	CHARACTER	160	CSB_TRACE_ENTRIES_START	
(1E0)	CHARACTER	16	CSB_TRACE_TABLE_ENTRY (10)	
(280)	CHARACTER	16	CSB_TRACE_TAIL	End of trace eycatchr@P1C
SDWA fields. The name and address fields may not always be available at the time of abend and will not contain correct info				
(290)	ADDRESS	4	CSB_SDWA_REGS (16)	SDWA registers 0-15
(2D0)	CHARACTER	8	CSB_SDWA_PSW	PSW at time of error
(2D8)	CHARACTER	8	CSB_SDWA_NAME	Abending program
(2E0)	ADDRESS	4	CSB_SDWA_ADDRESS	Abending prog addr

DFHD2IDT block (indoubt thread list)

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	DFHD2IDT	
(0)	CHARACTER	16	IDT_PREFIX	standard prefix
(0)	HALFWORD	2	IDT_LENGTH	
(2)	CHARACTER	14	IDT_EYE	>DFHD2IDT
(10)	HALFWORD	2	IDT_COUNT	number of indoubts
(12)	CHARACTER	20	IDT_ENTRY (*)	
(12)	CHARACTER	16	IDT_URID	UR ID (NID)
(22)	CHARACTER	4	IDT_DISPOSITION	disp of nid from show SHOW: nid is indoubt COMM: nid is a redo

Trace table entry dsect

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	DFHD2TR	
(0)	UNSIGNED	4	CSB_TRACE_REQUEST_NUM	request number
(4)	CHARACTER	4	CSB_TRACE_ REQUEST	request type
(8)	UNSIGNED	2	*	reserved
(A)	UNSIGNED	2	CSB_TRACE_FRBRC1	frb return code
(C)	UNSIGNED	4	CSB_TRACE_FRBRC2	frb reason code

Constants

Len	Type	Value	Name	Description
14	CHARACTER	>DFHD2CSB	DFHD2CSB_ EYECATCHER	
16	CHARACTER	>>Trace Start >>	CSB_TRACE_HEAD_EYE	
16	CHARACTER	<<Trace End <<	CSB_TRACE_TAIL_EYE	

D2ENT Db2entry block

CONTROL BLOCK NAME = DFHD2ENT
DESCRIPTIVE NAME = CICS DB2 attach DB2ENTRY control block
FUNCTION =
  The DFHD2ENT block represents a DB2ENTRY RDO object and
  holds state data and attributes to be used a transation
  or set of transactions when accessing DB2.
LIFETIME =
  A DFHD2ENT is getmained when a DB2ENTRY entity is installed.
  It is freemained when a DB2ENTRY is discarded.
LOCATION =
  DFHD2ENT resides above the 16MB line. It is located using
  Directory manager domain using its name as the key.
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS = none
 MODULE TYPE = Control block definition
  DFHD2ENT block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	200	DFHD2ENT	
(0)	STRUCTURE IsA(DFHD2RCT)	200	ENT	
(0)	CHARACTER	16	RCT_PREFIX	Standard Prefix
(0)	HALFWORD	2	RCT_LEN	
(2)	CHARACTER	14	RCT_EYE	
(10)	CHARACTER	8	RCT_NAME	DB2ENTRY name/POOL/COMD
(18)	CHARACTER	8	RCT_TIME	RCT time of install
(20)	CHARACTER	8	RCT_PLAN	Plan name if specified
(28)	CHARACTER	8	RCT_PLANEXIT_ NAME	Planexit name if specified
(30)	CHARACTER	4	RCT_TRANSID	Specified transid (if any)
(34)	ADDRESS	4	RCT_CSUB_ ADDRESS	Locates CSUB
(38)	CHARACTER	8	RCT_AUTHID	Authid if used
(40)	BITSTRING	1	RCT_AUTHTYPE	Authtype if used
	1...		RCT_AUTHTYPE_ GROUP	authtype=group
	.1..		RCT_AUTHTYPE_ SIGNID	authtype=signid
	..1.		RCT_AUTHTYPE_ TERM	authtype=term
	...1		RCT_AUTHTYPE_ TXID	authtype=txid
	.... 1...		RCT_AUTHTYPE_ OPID	authtype=opid
	.... .1..		RCT_AUTHTYPE_ USERID	authtype=userid
	11		*	reserved
(41)	BITSTRING	1	RCT_ACCOUNTREC	DB2 accounting to be done
	1...		RCT_ACCOUNT_ PER_UOW	account per UOW
	.1..		RCT_ACCOUNT_ PER_TASK	account per task
	..1.		RCT_ACCOUNT_ PER_TXID	account per transid change
	...1		RCT_ACCOUNT_ NONE	no accounting
	 1111		*	reserved
(42)	BITSTRING	1	RCT_DROLLBACK	Deadlock rollback
	1...		RCT_DROLLBACK_ YES	Drollback(yes)
	.111 1111		*	reserved
(43)	BITSTRING	1	RCT_PRIORITY	Priority of entry threads
	1...		RCT_PRIORITY_ HIGH	Higher than CICS TCB
	.1..		RCT_PRIORITY_ EQUAL	

Offset Hex	Type	Len	Name (Dim)	Description
				Equal to CICS TCB
	..1.		RCT_PRIORITY_ LOW	Lower than CICS TCB
	...1 1111		*	reserved
(44)	BITSTRING	1	RCT_THREADWAIT	Entry Threadwait setting
	1...		RCT_THREADWAIT_ YES	Wait for a thread
	.1..		RCT_THREADWAIT_ NO	Do not wait, abend
	..1.		RCT_THREADWAIT_ POOL	Overflow to the pool
	...1 1111		*	reserved
(45)	BITSTRING	1	RCT_ENABLED_ STATUS	Enable status of DB2ENTRY
	1...		RCT_DISABLED	DB2ENTRY is disabled
	.1..		RCT_DISABLING	DB2ENTRY is disabling
	..1.		RCT_DISABLED_ ROUTE_TO_POOL	Route new trans to pool
	...1		RCT_DISABLED_ BAD_SQLCODE	give new trans a sqlcode
	.... 1...		RCT_DISABLED_ ABEND_TRANS	abend new transactions
	.... .111		*	reserved
(46)	BITSTRING	2	*	reserved
(48)	CHARACTER	8	RCT_TAMPER_ CHECK1	check for overwrite
(50)	CHARACTER	8	RCT_TAMPER_ CHECK2	check for overwrite
(58)	UNSIGNED	4	RCT_THREAD_ LIMIT	Maximum active threads
(5C)	UNSIGNED	4	RCT_MAX_ PROTECTED_THREADS	Maximum protected threads
(60)	CHARACTER	8	RCT_THREADS	
(60)	UNSIGNED	4	RCT_CURRENT_ ACTIVE_THREADS	No of threads active
(64)	UNSIGNED	4	RCT_THREAD_HWM	hwm of active threads
(68)	CHARACTER	8	RCT_PROTECTED_ THREADS	
(68)	UNSIGNED	4	RCT_CURRENT_ PROTECTED_THREADS	No of prot. threads
(6C)	UNSIGNED	4	RCT_PROTECTED_ THREADS_HWM	hwm of protected threads
(70)	CHARACTER	8	RCT_USERS	
(70)	UNSIGNED	4	RCT_USE_COUNT	No. of tasks using entry
(74)	UNSIGNED	4	RCT_USE_ COUNT_HWM	hwm of tasks
(78)	CHARACTER	8	RCT_WAITERS	
(78)	UNSIGNED	4	RCT_READYQ_ COUNT	No. of tasks on readyq
(7C)	UNSIGNED	4	RCT_READYQ_HWM	hwm of tasks on readyq
(80)	UNSIGNED	4	RCT_TASK_COUNT	# tasks
(84)	UNSIGNED	4	RCT_CALL_COUNT	# calls
(88)	UNSIGNED	4	RCT_AUTH_COUNT	# authorisations
(8C)	UNSIGNED	4	RCT_COMMIT_ COUNT	# commits
(90)	UNSIGNED	4	RCT_ABORT_COUNT	# aborts
(94)	UNSIGNED	4	RCT_SINGLE_ PHASE_COUNT	# R/O commits & single up
(98)	UNSIGNED	4	RCT_THREAD_ REUSE_COUNT	# thread reuses
(9C)	UNSIGNED	4	RCT_THREAD_ TERM_COUNT	# thread terminates
(A0)	UNSIGNED	4	RCT_WAIT_ OR_OVERFLOW	# waits/overflow
(A4)	CHARACTER	4	RCT_DISABLE_ AREA	
(A4)	BITSTRING	1	RCT_DISABLE_ ECB	ECB for disabling
(A5)	UNSIGNED	3	RCT_DISABLE_ WAIT_COUNT	Count of waiters
(A8)	ADDRESS	4	RCT_DYNAMIC_ PLAN_EXIT_ANCHOR	Anchor for user area
CSUB chains				
(AC)	ADDRESS	4	RCT_ACTIVE_ THREAD_CHAIN	Active threads chain
(B0)	ADDRESS	4	RCT_FREE_ PROT_THREAD_CHAIN	Free protected threads

Offset Hex	Type	Len	Name (Dim)	Description
(B4)	ADDRESS	4	RCT_FREE_ TCB_CHAIN	Free TCBs for this entry
LOT Chain				
(B8)	ADDRESS	4	RCT_LOT_CHAIN	Chain of LOTs using entry
(BC)	ADDRESS	4	*	reserved to dword align
Readyq LOT chain.				
(C0)	CHARACTER	8	RCT_READYQ	
(C0)	ADDRESS	4	RCT_READYQ_ LOT_CHAIN	Readyq chain of LOTs
(C4)	UNSIGNED	4	RCT_READYQ_ SEC_COUNT	Security count for CDS

DFHD2RCT declares the whole of the layout of a DB2ENTRY as a type. The type is for the layout of the DB2ENTRY and for the layout of the pool and command sections in DFHD2GLB. Some fields, although declared, will not be used in the pool and command sections of DFHD2GLB.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	200	DFHD2RCT	
(0)	CHARACTER	16	RCT_PREFIX	Standard Prefix
(0)	HALFWORD	2	RCT_LEN	
(2)	CHARACTER	14	RCT_EYE	
(10)	CHARACTER	8	RCT_NAME	DB2ENTRY name/POOL/COMD
(18)	CHARACTER	8	RCT_TIME	RCT time of install
(20)	CHARACTER	8	RCT_PLAN	Plan name if specified
(28)	CHARACTER	8	RCT_PLANEXIT_ NAME	Planexit name if specified
(30)	CHARACTER	4	RCT_TRANSID	Specified transid (if any)
(34)	ADDRESS	4	RCT_CSUB_ADDRESS	Locates CSUB
(38)	CHARACTER	8	RCT_AUTHID	Authid if used
(40)	BITSTRING	1	RCT_AUTHTYPE	Authtype if used
	1...		RCT_AUTHTYPE_ GROUP	authtype=group
	.1..		RCT_AUTHTYPE_ SIGNID	authtype=signid
	..1.		RCT_AUTHTYPE_ TERM	authtype=term
	...1		RCT_AUTHTYPE_ TXID	authtype=txid
	.... 1...		RCT_AUTHTYPE_ OPID	authtype=opid
	.... .1..		RCT_AUTHTYPE_ USERID	authtype=userid
	11		*	reserved
(41)	BITSTRING	1	RCT_ACCOUNTREC	DB2 accounting to be done
	1...		RCT_ACCOUNT_ PER_UOW	account per UOW
	.1..		RCT_ACCOUNT_ PER_TASK	account per task
	..1.		RCT_ACCOUNT_ PER_TXID	account per transid change
	...1		RCT_ACCOUNT_ NONE	no accounting
	 1111		*	reserved
(42)	BITSTRING	1	RCT_DROLLBACK	Deadlock rollback
	1...		RCT_DROLLBACK_ YES	Drollback(yes)
	.111 1111		*	reserved
(43)	BITSTRING	1	RCT_PRIORITY	Priority of entry threads
	1...		RCT_PRIORITY_ HIGH	Higher than CICS TCB
	.1..		RCT_PRIORITY_ EQUAL	Equal to CICS TCB
	..1.		RCT_PRIORITY_ LOW	Lower than CICS TCB
	...1 1111		*	reserved
(44)	BITSTRING	1	RCT_THREADWAIT	Entry Threadwait setting
	1...		RCT_THREADWAIT_ YES	Wait for a thread
	.1..		RCT_THREADWAIT_ NO	Do not wait, abend
	..1.		RCT_THREADWAIT_ POOL	Overflow to the pool
	...1 1111		*	reserved
(45)	BITSTRING	1	RCT_ENABLED_ STATUS	Enable status of DB2ENTRY
	1...		RCT_DISABLED	DB2ENTRY is disabled
	.1..		RCT_DISABLING	DB2ENTRY is disabling
	..1.		RCT_DISABLED_ ROUTE_TO_POOL	Route new trans to pool

Offset Hex	Type	Len	Name (Dim)	Description
	...1		RCT_DISABLED_BAD_SQLCODE	give new trans a sqlcode
	.... 1...		RCT_DISABLED_ABEND_TRANS	abend new transactions
	.... .111		*	reserved
(46)	BITSTRING	2	*	reserved
(48)	CHARACTER	8	RCT_TAMPER_ CHECK1	check for overwrite
(50)	CHARACTER	8	RCT_TAMPER_ CHECK2	check for overwrite
(58)	UNSIGNED	4	RCT_THREAD_LIMIT	Maximum active threads
(5C)	UNSIGNED	4	RCT_MAX_PROTECTED_THREADS	Maximum protected threads
(60)	CHARACTER	8	RCT_THREADS	
(60)	UNSIGNED	4	RCT_CURRENT_ACTIVE_THREADS	No of threads active
(64)	UNSIGNED	4	RCT_THREAD_HWM	hwm of active threads
(68)	CHARACTER	8	RCT_PROTECTED_THREADS	
(68)	UNSIGNED	4	RCT_CURRENT_PROTECTED_THREADS	No of prot. threads
(6C)	UNSIGNED	4	RCT_PROTECTED_THREADS_HWM	hwm of protected threads
(70)	CHARACTER	8	RCT_USERS	
(70)	UNSIGNED	4	RCT_USE_COUNT	No. of tasks using entry
(74)	UNSIGNED	4	RCT_USE_ COUNT_HWM	hwm of tasks
(78)	CHARACTER	8	RCT_WAITERS	
(78)	UNSIGNED	4	RCT_READYQ_ COUNT	No. of tasks on readyq
(7C)	UNSIGNED	4	RCT_READYQ_HWM	hwm of tasks on readyq
(80)	UNSIGNED	4	RCT_TASK_COUNT	# tasks
(84)	UNSIGNED	4	RCT_CALL_COUNT	# calls
(88)	UNSIGNED	4	RCT_AUTH_COUNT	# authorisations
(8C)	UNSIGNED	4	RCT_COMMIT_COUNT	# commits
(90)	UNSIGNED	4	RCT_ABORT_COUNT	# aborts
(94)	UNSIGNED	4	RCT_SINGLE_PHASE_COUNT	# R/O commits & single up
(98)	UNSIGNED	4	RCT_THREAD_REUSE_COUNT	# thread reuses
(9C)	UNSIGNED	4	RCT_THREAD_TERM_COUNT	# thread terminates
(A0)	UNSIGNED	4	RCT_WAIT_OR_OVERFLOW	# waits/overflow
(A4)	CHARACTER	4	RCT_DISABLE_AREA	
(A4)	BITSTRING	1	RCT_DISABLE_ECB	ECB for disabling
(A5)	UNSIGNED	3	RCT_DISABLE_WAIT_COUNT	Count of waiters
(A8)	ADDRESS	4	RCT_DYNAMIC_PLAN_EXIT_ANCHOR	Anchor for user area
(AC)	ADDRESS	4	RCT_ACTIVE_THREAD_CHAIN	Active threads chain
(B0)	ADDRESS	4	RCT_FREE_PROT_THREAD_CHAIN	Free protected threads
(B4)	ADDRESS	4	RCT_FREE_ TCB_CHAIN	Free TCBs for this entry
(B8)	ADDRESS	4	RCT_LOT_CHAIN	Chain of LOTs using entry
(BC)	ADDRESS	4	*	reserved to dword align
(C0)	CHARACTER	8	RCT_READYQ	
(C0)	ADDRESS	4	RCT_READYQ_LOT_CHAIN	Readyq chain of LOTs
(C4)	UNSIGNED	4	RCT_READYQ_SEC_COUNT	Security count for CDS

Constants

Len	Type	Value	Name	Description
14	CHARACTER	>DFHD2ENT	DFHD2ENT_ EYECATCHER	

D2GLB Cics/db2 global block

CONTROL BLOCK NAME = DFHD2GLB
DESCRIPTIVE NAME = CICS DB2 attach Global block
FUNCTION =
The DFHD2GLB block represents the DB2CONN RDO object and contains global state information for the CICS-DB2 connection. It also contains the state information for
Pool threads and command threads. These are mapped by the generic DB2ENTRY structure DFHD2RCT but are included in the DFHD2GLB as there can only be one pool definition and command thread definition and hence are global in nature.
A DB2CONN and hence a DFHD2GLB is the minimum required to operate the CICS-DB2 Attachment facility.
LIFETIME =
A DFHD2GLB is getmained when a DB2CONN entity is installed.
It is freemained when a DB2CONN is discarded.
LOCATION =
DFHD2GLB is anchored off CICS/DB2 static storage (DFHD2SS).
It resides above the 16MB line.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = none
MODULE TYPE = Control block definition
DFHD2GLB block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1360	DFHD2GLB	
(0)	CHARACTER	16	GLB_PREFIX	Standard Prefix
(0)	HALFWORD	2	GLB_LEN	
(2)	CHARACTER	14	GLB_EYE	>DFHD2GLB
Global information				
(10)	CHARACTER	8	GLB_DB2CONN_NAME	Name of DB2CONN
(18)	CHARACTER	8	GLB_CICS_ID	Name of CICS
(20)	CHARACTER	4	GLB_DB2_ID	Name of DB2
(24)	CHARACTER	4	GLB_DB2_RELEASE	Release of DB2
(28)	ADDRESS	4	GLB_DSNAPRH_ ENTRY	Entry point of DSNAPRH
(2C)	ADDRESS	4	GLB_DFHD2EX1_GWA_ADDR	Address of GWA of EX1
(30)	ADDRESS	4	GLB_DFHD2EX2_ ENTRY	Entry point of DFHD2EX2
(34)	ADDRESS	4	GLB_DFHD2EX3_ ENTRY	Entry Point of DFHD2EX3
(38)	ADDRESS	4	GLB_DFHD2MSB_ ENTRY	Entry point of DFHD2MSB
(3C)	ADDRESS	4	GLB_MSB_TCB	DFHD2MSB tcb address
(40)	CHARACTER	12	GLB_MSG_QUEUES	Message destinations
(40)	CHARACTER	4	GLB_MSG_QUEUE1	Message destination 1
(44)	CHARACTER	4	GLB_MSG_QUEUE2	Message destination 2
(48)	CHARACTER	4	GLB_MSG_QUEUE3	Message destination 3
(4C)	CHARACTER	8	GLB_SIGNON_ID	Id for authtype(signid)
(54)	CHARACTER	8	GLB_SECURITY_REBUILD_TIME	STCK for security rebuild
(5C)	CHARACTER	8	GLB_CONNECT_TIME	STCK when connected
(64)	CHARACTER	8	GLB_DISCONNECT_ TIME	STCK when disconnected
(6C)	CHARACTER	4	GLB_STATS_QUEUE	Statistics destination
(70)	CHARACTER	8	GLB_PURGE_CYCLE	Prot. Thread purge cycle
(70)	UNSIGNED	4	GLB_PURGE_CYCLE_MINUTES	Purge cycle minutes
(74)	UNSIGNED	4	GLB_PURGE_CYCLE_SECONDS	Purge cycle seconds
(78)	CHARACTER	8	GLB_TCBS	
(78)	UNSIGNED	4	GLB_CURRENT_ TCBS	Current TCB number
(7C)	UNSIGNED	4	GLB_TCB_HWM	hwm of tcbs
(80)	UNSIGNED	4	GLB_TCB_LIMIT	Max number of TCBs
(84)	ADDRESS	4	GLB_FREE_ TCB_CHAIN	Global Free TCB chain
(88)	UNSIGNED	4	GLB_FREE_ TCB_COUNT	Number of free TCBs
(8C)	UNSIGNED	4	*	reserved
(90)	CHARACTER	8	GLB_TCB_READYQ	

Offset Hex	Type	Len	Name (Dim)	Description
(90)	ADDRESS	4	GLB_TCB_READYQ_CHAIN	Readyq for free TCBs
(94)	ADDRESS	4	GLB_TCB_READYQ_SEC_COUNT	Security count for CDS
(98)	CHARACTER	8	GLB_TCB_READYQ_COUNTS	
(98)	UNSIGNED	4	GLB_TCB_READYQ_COUNT	Number of tasks on readyq
(9C)	UNSIGNED	4	GLB_TCB_ READYQ_HWM	Peak no. of tasks
(A0)	ADDRESS	4	GLB_FREE_PROT_THREAD_ CHAIN1	Global Free Prot.threads
(A4)	ADDRESS	4	GLB_FREE_PROT_THREAD_ CHAIN2	Global Free Prot.threads
(A8)	BITSTRING	1	GLB_FLAGS	DB2CONN state flags
	1...		GLB_DISCARDING_DB2CONN	Discard in progress
	.111 1111		*	Reserved
(A9)	BITSTRING	1	GLB_STANDBY_MODE	Standby mode
	1...		GLB_STANDBY_MODE_RECONNECT	Standby=reconnect
	.1..		GLB_STANDBY_MODE_CONNECT	Standby=connect
	..1.		GLB_STANDBY_MODE_NOCONNECT	Standby=noconnect
	...1 1111		*	Reserved
(AA)	BITSTRING	1	GLB_CONNECT_ ERROR	Connect error action
	1...		GLB_CONNECT_ERROR_SQLCODE	Connecterror=sqlcode
	.1..		GLB_CONNECT_ERROR_ABEND	Connecterror=abend
	..11 1111		*	Reserved
(AB)	BITSTRING	1	GLB_NON_TERMINAL_RELEASE	Nontermrel attribute
	1...		GLB_NON_ TERMINAL_RELEASE_YES	Nontermrel=yes
	.111 1111		*	Reserved
(AC)	BITSTRING	1	GLB_THREAD_ERROR	Thread error action
	1...		GLB_THREAD_ERROR_ABEND	Threaderror=abend
	.1..		GLB_THREAD_ERROR_N906D	Threaderror=n906d
	..1.		GLB_THREAD_ERROR_N906	Threaderror=n906
	...1 1111		*	Reserved
(AD)	BITSTRING	1	GLB_CONNECTION_STATUS	CICS-DB2 Connection state
	1...		GLB_CONNECTED	Connected to DB2
	.1..		GLB_CONNECTING	Connecting to DB2
	..1.		GLB_DISCONNECTING	Disconnecting from DB2
	...1 1111		*	Reserved
(AE)	BITSTRING	1	GLB_ATTACH_ STATUS	CICS Attachment status
	1...		GLB_IN_STANDBY	Attach is in standby
	.1..		GLB_SERVICE_TASK_STARTED	CEX2 has started
	..1.		GLB_SERVICE_TASK_TERMINATE	CEX2 should terminate
	...1		GLB_DB2_ ACCMAINT	DB2 is in access(maint)
	.... 1...		GLB_DFHD2MSB_ ACTIVE	DFHD2MSB is active
	.... .111		*	reserved
(AF)	BITSTRING	1	GLB_SHUTDOWN_ FLAGS	shutdown flags
	1...		GLB_SHUTDOWN_QUIESCE	quiesce shutdown
	.1..		GLB_SHUTDOWN_ FORCE	force shutdown

Offset Hex	Type	Len	Name (Dim)	Description
	..1.		GLB_SHUTDOWN_ DB2	shutdown initiated by DB2
	...1		GLB_SHUTDOWN_ MSB_ESTAE	shutdown due to DFHD2MSB abending
	.... 1...		GLB_SHUTDOWN_ CICS_IMMED	shutdown due to immediate shutdown of CICS.
	.... .1..		GLB_SHUTDOWN_ CICS_QUIESCE	shutdown due to quiesce shutdown of CICS
	.... ..1.		GLB_SHUTDOWN_ EX2	shutdown initiated by service task CEX2
	1		GLB_SHUTDOWN_ EX1_FINAL	call is from EX1 to complete shutdown
(B0)	UNSIGNED	4	GLB_SERVICE_ TASK_ECB	request for service ECB
(B4)	UNSIGNED	4	GLB_SERVICE_ TASK_STOP_ECB	wait for CEX2 to term
(B8)	FULLWORD	4	*	reserved
(BC)	UNSIGNED	4	GLB_SERVICE_ TASK_P_COUNT	number of purges by EX2
(C0)	CHARACTER	428	GLB_MSB_AREA	DFHD2MSB storage area
(C0)	ADDRESS	4	GLB_INDOUBT_ LIST	resync indoubt list
(C4)	UNSIGNED	2	GLB_INDOUBTS_ LENGTH	resync indoubts length
(C6)	UNSIGNED	2	GLB_INDOUBTS_ COUNT	resync indoubts count
(C8)	ADDRESS	4	GLB_UR_ INDOUBT_LOT_ADDR	UR indoubt LOT chain
(CC)	ADDRESS	4	GLB_ATTACH_ DETACH_CHAIN	Global attach/Detach chn
(D0)	UNSIGNED	4	GLB_MSB_ WAIT_ECB	main task wait ECB
(D4)	UNSIGNED	4	GLB_MSB_ LISTEN_ECB	main subtask listen ecb
(D8)	FULLWORD	4	GLB_MSB_ START_ECB	strt ecb for start comp.
(DC)	UNSIGNED	4	GLB_MSB_ STOP_ECB	main task wait purge ECB
(E0)	BITSTRING	1	GLB_MSB_PARM4	savearea for estae rc
(E1)	BITSTRING	1	GLB_MSB_PARM3	D2MSB error flags
	1...		GLB_MSB_ LOAD_PRH_FAILED	failed to load prh
	.1..		GLB_MSB_ DB2_NOT_ACTIVE	db2 is not active
	..1.		GLB_MSB_ DB2_IDENTIFY_ FAILED	identify to DB2 failed
	...1		GLB_MSB_ INSUFFICIENT_AUTH	auth identify failed
	.... 1...		GLB_MSB_ ABENDING	D2MSB is abending
	.... .1..		GLB_MSB_ SHOW_INDOUBT_ FAILED	show indoubt failed
	.... ..1.		GLB_MSB_ EST_ESTAE_FAILED	Failed to establish estae
	1		GLB_MSB_ EST_EXIT_FAILED	Failed to estab.SSSC exit
(E2)	BITSTRING	1	GLB_MSB_PARM2	identify flags
	1...		GLB_TERMINATE_ IDENTIFY	terminate identify
	.1..		GLB_IDENTIFY_ TERMINATED	identify terminated
	..1.		GLB_CICS_ CHAPPED_DOWN	CICS priority lowered
	...1		GLB_MSB_ ISSUED_ABEND	Abend requested
	 1111		*	Reserved
(E3)	BITSTRING	1	*	reserved
(E4)	CHARACTER	72	GLB_MSB_ SAVEAREA	DFHD2MSB fwd save area
(12C)	CHARACTER	72	GLB_ATTACH_ PARMLIST	attach parameter list
(174)	CHARACTER	200	GLB_WORKAREA	workarea
(23C)	CHARACTER	48	GLB_FRB	space for glb FRB
(26C)	CHARACTER	252	GLB_THREAD_ NUMBERS	Bitmap for CSUB nums

Offset Hex	Type	Len	Name (Dim)	Description
(26C)	ADDRESS	4	GLB_THREAD_ NUM_WORDS (63)	
(368)	ADDRESS	4	GLB_STATS_ BUFFER_ADDR	Address of stats buffer
SDWA fields. The name and address fields may not always be available at the time of abend and will not contain correct info				
(36C)	ADDRESS	4	GLB_SDWA_REGS (16)	SDWA reg 0-15
(3AC)	CHARACTER	8	GLB_SDWA_PSW	PSW at error time
(3B4)	CHARACTER	8	GLB_SDWA_NAME	Abending prog name
(3BC)	ADDRESS	4	GLB_SDWA_ADDRESS	Abending prog addr
Pool threads section				
(3C0)	STRUCTURE IsA(DFHD2RCT)	200	GLB_POOL	Double word aligned
(3C0)	CHARACTER	16	RCT_PREFIX	Standard Prefix
(3C0)	HALFWORD	2	RCT_LEN	
(3C2)	CHARACTER	14	RCT_EYE	
(3D0)	CHARACTER	8	RCT_NAME	DB2ENTRY name/POOL/COMD
(3D8)	CHARACTER	8	RCT_TIME	RCT time of install
(3E0)	CHARACTER	8	RCT_PLAN	Plan name if specified
(3E8)	CHARACTER	8	RCT_PLANEXIT_ NAME	Planexit name if specified
(3F0)	CHARACTER	4	RCT_TRANSID	Specified transid (if any)
(3F4)	ADDRESS	4	RCT_CSUB_ ADDRESS	Locates CSUB
(3F8)	CHARACTER	8	RCT_AUTHID	Authid if used
(400)	BITSTRING	1	RCT_AUTHTYPE	Authtype if used
	1...		RCT_AUTHTYPE_ GROUP	authtype=group
	.1..		RCT_AUTHTYPE_ SIGNID	authtype=signid
	..1.		RCT_AUTHTYPE_ TERM	authtype=term
	...1		RCT_AUTHTYPE_ TXID	authtype=txid
	.... 1...		RCT_AUTHTYPE_ OPID	authtype=opid
	.... .1..		RCT_AUTHTYPE_ USERID	authtype=userid
	11		*	reserved
(401)	BITSTRING	1	RCT_ACCOUNTREC	DB2 accounting to be done
	1...		RCT_ACCOUNT_ PER_UOW	account per UOW
	.1..		RCT_ACCOUNT_ PER_TASK	account per task
	..1.		RCT_ACCOUNT_ PER_TXID	account per transid change
	...1		RCT_ACCOUNT_ NONE	no accounting
	 1111		*	reserved
(402)	BITSTRING	1	RCT_DROLLBACK	Deadlock rollback
	1...		RCT_DROLLBACK_ YES	Drollback(yes)
	.111 1111		*	reserved
(403)	BITSTRING	1	RCT_PRIORITY	Priority of entry threads
	1...		RCT_PRIORITY_ HIGH	Higher than CICS TCB
	.1..		RCT_PRIORITY_ EQUAL	Equal to CICS TCB
	..1.		RCT_PRIORITY_ LOW	Lower than CICS TCB
	...1 1111		*	reserved
(404)	BITSTRING	1	RCT_THREADWAIT	Entry Threadwait setting
	1...		RCT_THREADWAIT_ YES	Wait for a thread
	.1..		RCT_THREADWAIT_ NO	Do not wait, abend
	..1.		RCT_THREADWAIT_ POOL	Overflow to the pool
	...1 1111		*	reserved
(405)	BITSTRING	1	RCT_ENABLED_ STATUS	Enable status of DB2ENTRY
	1...		RCT_DISABLED	DB2ENTRY is disabled
	.1..		RCT_DISABLING	DB2ENTRY is disabling
	..1.		RCT_DISABLED_ ROUTE_TO_POOL	Route new trans to pool
	...1		RCT_DISABLED_ BAD_SQLCODE	give new trans a sqlcode
	.... 1...		RCT_DISABLED_ ABEND_TRANS	

Offset Hex	Type	Len	Name (Dim)	Description
				abend new transactions
	.... .111		*	reserved
(406)	BITSTRING	2	*	reserved
(408)	CHARACTER	8	RCT_TAMPER_ CHECK1	check for overwrite
(410)	CHARACTER	8	RCT_TAMPER_ CHECK2	check for overwrite
(418)	UNSIGNED	4	RCT_THREAD_ LIMIT	Maximum active threads
(41C)	UNSIGNED	4	RCT_MAX_PROTECTED_THREADS	Maximum protected threads
(420)	CHARACTER	8	RCT_THREADS	
(420)	UNSIGNED	4	RCT_CURRENT_ACTIVE_THREADS	No of threads active
(424)	UNSIGNED	4	RCT_THREAD_HWM	hwm of active threads
(428)	CHARACTER	8	RCT_PROTECTED_THREADS	
(428)	UNSIGNED	4	RCT_CURRENT_PROTECTED_THREADS	No of prot. threads
(42C)	UNSIGNED	4	RCT_PROTECTED_THREADS_HWM	hwm of protected threads
(430)	CHARACTER	8	RCT_USERS	
(430)	UNSIGNED	4	RCT_USE_COUNT	No. of tasks using entry
(434)	UNSIGNED	4	RCT_USE_ COUNT_HWM	hwm of tasks
(438)	CHARACTER	8	RCT_WAITERS	
(438)	UNSIGNED	4	RCT_READYQ_ COUNT	No. of tasks on readyq
(43C)	UNSIGNED	4	RCT_READYQ_HWM	hwm of tasks on readyq
(440)	UNSIGNED	4	RCT_TASK_COUNT	# tasks
(444)	UNSIGNED	4	RCT_CALL_COUNT	# calls
(448)	UNSIGNED	4	RCT_AUTH_COUNT	# authorisations
(44C)	UNSIGNED	4	RCT_COMMIT_ COUNT	# commits
(450)	UNSIGNED	4	RCT_ABORT_COUNT	# aborts
(454)	UNSIGNED	4	RCT_SINGLE_PHASE_COUNT	# R/O commits & single up
(458)	UNSIGNED	4	RCT_THREAD_REUSE_COUNT	# thread reuses
(45C)	UNSIGNED	4	RCT_THREAD_TERM_COUNT	# thread terminates
(460)	UNSIGNED	4	RCT_WAIT_OR_OVERFLOW	# waits/overflow
(464)	CHARACTER	4	RCT_DISABLE_ AREA	
(464)	BITSTRING	1	RCT_DISABLE_ ECB	ECB for disabling
(465)	UNSIGNED	3	RCT_DISABLE_WAIT_COUNT	Count of waiters
(468)	ADDRESS	4	RCT_DYNAMIC_PLAN_EXIT_ANCHOR	Anchor for user area
CSUB chains				
(46C)	ADDRESS	4	RCT_ACTIVE_THREAD_CHAIN	Active threads chain
(470)	ADDRESS	4	RCT_FREE_PROT_THREAD_CHAIN	Free protected threads
(474)	ADDRESS	4	RCT_FREE_ TCB_CHAIN	Free TCBs for this entry
LOT Chain				
(478)	ADDRESS	4	RCT_LOT_CHAIN	Chain of LOTs using entry
(47C)	ADDRESS	4	*	reserved to dword align
Readyq LOT chain.				
(480)	CHARACTER	8	RCT_READYQ	
(480)	ADDRESS	4	RCT_READYQ_LOT_CHAIN	Readyq chain of LOTs
(484)	UNSIGNED	4	RCT_READYQ_SEC_COUNT	Security count for CDS
Command threads section				
(488)	STRUCTURE IsA(DFHD2RCT)	200	GLB_COMD	
(488)	CHARACTER	16	RCT_PREFIX	Standard Prefix
(488)	HALFWORD	2	RCT_LEN	
(48A)	CHARACTER	14	RCT_EYE	
(498)	CHARACTER	8	RCT_NAME	DB2ENTRY name/POOL/COMD
(4A0)	CHARACTER	8	RCT_TIME	RCT time of install

Offset Hex	Type	Len	Name (Dim)	Description
(4A8)	CHARACTER	8	RCT_PLAN	Plan name if specified
(4B0)	CHARACTER	8	RCT_PLANEXIT_ NAME	Planexit name if specified
(4B8)	CHARACTER	4	RCT_TRANSID	Specified transid (if any)
(4BC)	ADDRESS	4	RCT_CSUB_ ADDRESS	Locates CSUB
(4C0)	CHARACTER	8	RCT_AUTHID	Authid if used
(4C8)	BITSTRING	1	RCT_AUTHTYPE	Authtype if used
	1...		RCT_AUTHTYPE_GROUP	authtype=group
	.1..		RCT_AUTHTYPE_ SIGNID	authtype=signid
	..1.		RCT_AUTHTYPE_ TERM	authtype=term
	...1		RCT_AUTHTYPE_ TXID	authtype=txid
	.... 1...		RCT_AUTHTYPE_ OPID	authtype=opid
	.... .1..		RCT_AUTHTYPE_USERID	authtype=userid
	11		*	reserved
(4C9)	BITSTRING	1	RCT_ACCOUNTREC	DB2 accounting to be done
	1...		RCT_ACCOUNT_PER_UOW	account per UOW
	.1..		RCT_ACCOUNT_PER_TASK	account per task
	..1.		RCT_ACCOUNT_PER_TXID	account per transid change
	...1		RCT_ACCOUNT_ NONE	no accounting
	 1111		*	reserved
(4CA)	BITSTRING	1	RCT_DROLLBACK	Deadlock rollback
	1...		RCT_DROLLBACK_ YES	Drollback(yes)
	.111 1111		*	reserved
(4CB)	BITSTRING	1	RCT_PRIORITY	Priority of entry threads
	1...		RCT_PRIORITY_ HIGH	Higher than CICS TCB
	.1..		RCT_PRIORITY_ EQUAL	Equal to CICS TCB
	..1.		RCT_PRIORITY_ LOW	Lower than CICS TCB
	...1 1111		*	reserved
(4CC)	BITSTRING	1	RCT_THREADWAIT	Entry Threadwait setting
	1...		RCT_THREADWAIT_ YES	Wait for a thread
	.1..		RCT_THREADWAIT_ NO	Do not wait, abend
	..1.		RCT_THREADWAIT_POOL	Overflow to the pool
	...1 1111		*	reserved
(4CD)	BITSTRING	1	RCT_ENABLED_ STATUS	Enable status of DB2ENTRY
	1...		RCT_DISABLED	DB2ENTRY is disabled
	.1..		RCT_DISABLING	DB2ENTRY is disabling
	..1.		RCT_DISABLED_ROUTE_TO_POOL	Route new trans to pool
	...1		RCT_DISABLED_BAD_SQLCODE	give new trans a sqlcode
	.... 1...		RCT_DISABLED_ABEND_TRANS	abend new transactions
	.... .111		*	reserved
(4CE)	BITSTRING	2	*	reserved
(4D0)	CHARACTER	8	RCT_TAMPER_ CHECK1	check for overwrite
(4D8)	CHARACTER	8	RCT_TAMPER_ CHECK2	check for overwrite
(4E0)	UNSIGNED	4	RCT_THREAD_ LIMIT	Maximum active threads
(4E4)	UNSIGNED	4	RCT_MAX_PROTECTED_THREADS	Maximum protected threads
(4E8)	CHARACTER	8	RCT_THREADS	
(4E8)	UNSIGNED	4	RCT_CURRENT_ACTIVE_THREADS	No of threads active
(4EC)	UNSIGNED	4	RCT_THREAD_HWM	hwm of active threads
(4F0)	CHARACTER	8	RCT_PROTECTED_THREADS	
(4F0)	UNSIGNED	4	RCT_CURRENT_PROTECTED_THREADS	No of prot. threads
(4F4)	UNSIGNED	4	RCT_PROTECTED_THREADS_HWM	

Offset Hex	Type	Len	Name (Dim)	Description
				hwm of protected threads
(4F8)	CHARACTER	8	RCT_USERS	
(4F8)	UNSIGNED	4	RCT_USE_COUNT	No. of tasks using entry
(4FC)	UNSIGNED	4	RCT_USE_ COUNT_HWM	hwm of tasks
(500)	CHARACTER	8	RCT_WAITERS	
(500)	UNSIGNED	4	RCT_READYQ_ COUNT	No. of tasks on readyq
(504)	UNSIGNED	4	RCT_READYQ_HWM	hwm of tasks on readyq
(508)	UNSIGNED	4	RCT_TASK_COUNT	# tasks
(50C)	UNSIGNED	4	RCT_CALL_COUNT	# calls
(510)	UNSIGNED	4	RCT_AUTH_COUNT	# authorisations
(514)	UNSIGNED	4	RCT_COMMIT_ COUNT	# commits
(518)	UNSIGNED	4	RCT_ABORT_COUNT	# aborts
(51C)	UNSIGNED	4	RCT_SINGLE_PHASE_COUNT	# R/O commits & single up
(520)	UNSIGNED	4	RCT_THREAD_REUSE_COUNT	# thread reuses
(524)	UNSIGNED	4	RCT_THREAD_TERM_COUNT	# thread terminates
(528)	UNSIGNED	4	RCT_WAIT_OR_OVERFLOW	# waits/overflow
(52C)	CHARACTER	4	RCT_DISABLE_ AREA	
(52C)	BITSTRING	1	RCT_DISABLE_ ECB	ECB for disabling
(52D)	UNSIGNED	3	RCT_DISABLE_WAIT_COUNT	Count of waiters
(530)	ADDRESS	4	RCT_DYNAMIC_PLAN_EXIT_ANCHOR	Anchor for user area
(534)	ADDRESS	4	RCT_ACTIVE_THREAD_CHAIN	Active threads chain
(538)	ADDRESS	4	RCT_FREE_PROT_THREAD_CHAIN	Free protected threads
(53C)	ADDRESS	4	RCT_FREE_ TCB_CHAIN	Free TCBs for this entry
(540)	ADDRESS	4	RCT_LOT_CHAIN	Chain of LOTs using entry
(544)	ADDRESS	4	*	reserved to dword align
(548)	CHARACTER	8	RCT_READYQ	
(548)	ADDRESS	4	RCT_READYQ_LOT_CHAIN	Readyq chain of LOTs
(54C)	UNSIGNED	4	RCT_READYQ_SEC_COUNT	Security count for CDS

Constants

Len	Type	Value	Name	Description
14	CHARACTER	>DFHD2GLB	DFHD2GLB_ EYECATCHER	
14	CHARACTER	GLB POOL SECTN	DFHD2GLB_POOL_EYECATCHER	
14	CHARACTER	GLB COMD SECTN	DFHD2GLB_COMD_EYECATCHER	
8	CHARACTER	*POOL	DFHD2GLB_POOL_NAME	
8	CHARACTER	*COMMAND	DFHD2GLB_COMD_NAME	
4	DECIMAL	14336	GLB_STATS_ BUFFER_LEN	

D2GWA Cics/db2 global work area

CONTROL BLOCK NAME = DFHD2GWA
DESCRIPTIVE NAME = CICS DB2 True's Global Work Area
FUNCTION =
  Global Work area for the CICS-DB2 True.
LIFETIME =
  The DFHD2GWA is getmained by CICS when the CICS-DB2 TRUE DFHD2EX1 is enabled. It is freemained when the TRUE is disabled.
LOCATION =
  DFHD2GWA resides below the 16MB line. It is located using UEPGAA in the TRUE's DFHUEPAR parameter list
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS = none
 MODULE TYPE = Control block definition
  DFHD2GWA

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	DFHD2GWA	
(0)	CHARACTER	8	GWA_PREFIX	Standard Prefix
(0)	HALFWORD	2	GWA_LENGTH	
(2)	CHARACTER	6	GWA_EYE	>D2GWA
(8)	ADDRESS	4	GWA_OLD_RCT	old RCT addr, must be at +8
(C)	ADDRESS	4	GWA_LOT	Chain of LOTs using DB2

Constants

Len	Type	Value	Name	Description
6	CHARACTER	>D2GWA	DFHD2GWA_ EYECATCHER	

D2LOT Cics/db2 life of task block

CONTROL BLOCK NAME = DFHD2LOT
DESCRIPTIVE NAME = CICS DB2 attach Life of Task block
FUNCTION =
The DFHD2LOT block holds task lifetime information about the task currently accessing DB2. It is the CICS-DB2 equivalent of the TCA.
LIFETIME =
The DFHD2LOT is a mapping of the task Local work area of the CICS-DB2 TRUE DFHD2EX1. It is getmained by CICS the time a CICS task calls the CICS-DB2 TRUE. It is freemained by CICS at task termination time.
LOCATION =
DFHD2LOT resides above the 16MB line. It is located using UEPTAA in the TRUE's DFHUEPAR parameter list
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = none
MODULE TYPE = Control block definition
DFHD2LOT

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	222	DFHD2LOT	
(0)	CHARACTER	16	LOT_PREFIX	Standard Prefix
(0)	HALFWORD	2	LOT_LEN	
(2)	CHARACTER	14	LOT_EYE	>DFHD2LOT
(10)	CHARACTER	4	LOT_TRANSID	Transaction id
(14)	ADDRESS	4	LOT_TCA	Address of TCA
(18)	ADDRESS	4	LOT_RCTE	-> DB2ENTRY | POOL | COMD
(1C)	ADDRESS	4	LOT_CSUB	Address of CSUB
(20)	ADDRESS	4	LOT_GWA_ CHAIN_NEXT	-> next LOT on GWA chain
(24)	ADDRESS	4	LOT_GWA_ CHAIN_PREV	-> prev LOT on GWA chain
(28)	ADDRESS	4	LOT_RCT_ CHAIN_NEXT	-> next LOT on RCT chain
(2C)	ADDRESS	4	LOT_RCT_ CHAIN_PREV	-> prev LOT on RCT chain
(30)	ADDRESS	4	LOT_CALL_PARMS	Addr of SQL or CICS parms
(30)	BITSTRING	1	*	
	1...		LOT_CALL_PARMS_HIGH	High bit of address
(34)	UNSIGNED	4	LOT_ECB	Ecb to wait CICS task
(38)	UNSIGNED	4	LOT_ACEE_ADDRESS	ACEE address
(3C)	UNSIGNED	4	LOT_WLM_ PERF_TOKEN	WLM performance token
(40)	CHARACTER	8	LOT_RCTE_READYQ	
(40)	ADDRESS	4	LOT_READYQ_NEXT	-> next LOT on readyq
(44)	UNSIGNED	4	LOT_READYQ_ COUNT	-> security count for CDS
(48)	CHARACTER	8	LOT_GLB_ TCB_READYQ	
(48)	ADDRESS	4	LOT_TCB_ READYQ_NEXT	-> next LOT on readyq
(4C)	UNSIGNED	4	LOT_TCB_READYQ_COUNT	-> security count for CDS
(50)	UNSIGNED	4	LOT_INDOUBT_NEXT	-> next LOT on indoubtq
(54)	CHARACTER	8	LOT_PLAN_NAME	Plan name
(5C)	CHARACTER	12	LOT_REQUEST_INDICATORS	
(5C)	BITSTRING	1	LOT_CURRENT_REQUEST	current request type
(5D)	BITSTRING	1	LOT_REQUEST_MINUS_ONE	previous request type
(5E)	BITSTRING	1	LOT_REQUEST_MINUS_TWO	current - 2 request type
(5F)	BITSTRING	1	LOT_REQUEST_MINUS_THREE	current - 3 request type
(60)	BITSTRING	1	LOT_REQUEST_ FLAGS	Miscellaneous flags
	1...		LOT_DYN_PLAN_ALLOWED	Allowed to call dyn plan
	.1..		LOT_APPL_MUST_ABORT	application must abort
	..1.		LOT_TERMINAL_ TRANS	terminal driven trans
	...1		LOT_OVERFLOW_TO_POOL	we have oveflowed to pool
	.... 1...		LOT_TXNS_ LAST_CALL	

Offset Hex	Type	Len	Name (Dim)	Description
				last uow for transaction
	.... .1..		LOT_TASK_PURGED_FROM_CICS	task purged from CICS
	11		*	reserved
(61)	BITSTRING	1	LOT_READ_ONLY_INDICATOR	read only commit ind.
	1...		LOT_PREPARE_READ_ONLY	prepare signalled r/o
	.111 1111		*	
(62)	BITSTRING	1	LOT_TRACE_FLAGS	copy of trace flags
	1...		LOT_LEVEL1_ TRACE	RMI level 1 trace active
	.1..		LOT_LEVEL2_ TRACE	RMI level 2 trace active
	..11 1111		*	reserved
(63)	BITSTRING	1	LOT_DEFERRED_ ABENDS	deferred abend flags
	1...		LOT_ABEND_AD2S	AD2S if more calls
	.1..		LOT_ABEND_AD2T	AD2T if more calls
	..1.		LOT_ABEND_AD2U	AD2U if more calls
	...1 1111		*	reserved
(64)	BITSTRING	4	LOT_SWAP_WORD	Word for compare & swap
(64)	BITSTRING	3	*	reserved
(67)	BITSTRING	1	LOT_SQL_STATUS	status of sql request
	1...		LOT_API_CALL_IN_PROGRESS	sql api call in progress
	.111 1111		LOT_API_ DETACH	All 8 bits set for detach
(68)	CHARACTER	8	LOT_RETURN_CODES	
(68)	UNSIGNED	1	LOT_RMI_ RETURN_CODE	Return code to CICS
(69)	BITSTRING	1	LOT_ERROR_CODES	General error code
(6A)	BITSTRING	1	LOT_ERROR_CODES_MINUS_ONE	error from previous req
(6B)	BITSTRING	1	LOT_ERROR_CODES_MINUS_TWO	error from req-2
(6C)	BITSTRING	1	LOT_ERROR_CODES_MINUS_THREE	error from req-3
(6D)	BITSTRING	3	*	reserved
(70)	CHARACTER	16	LOT_UR_TOKEN	UR token
(80)	CHARACTER	8	LOT_PRIMARY_AUTH_NAME	Auth name to sign on
(88)	CHARACTER	8	LOT_SECONDARY_AUTH_NAME	Secondary auth to sign on
(90)	CHARACTER	8	LOT_SUBTASK_ABEND_REASON	reason code and abend if subtask abended
(98)	CHARACTER	22	LOT_ACCOUNTING_ TOKEN	Accounting token for DB2
(98)	CHARACTER	8	LOT_ACCOUNT_NETNAME	Netname
(A0)	CHARACTER	8	LOT_ACCOUNT_ LUNAME	luname
(A8)	CHARACTER	6	LOT_ACCOUNT_ CLOCK	middle six bytes of STCK
(AE)	CHARACTER	48	LOT_FRB	space for clot FRB

Constants

Len	Type	Value	Name	Description
14	CHARACTER	>DFHD2LOT	DFHD2LOT_ EYECATCHER	
Constants for LOT_ current_request				
1	HEX	00	LOT_UR_SHOULD_NOT_BE_INDOUBT	
1	HEX	01	LOT_SQL_API_ REQUEST	
1	HEX	02	LOT_PREPARE_ REQUEST	
1	HEX	03	LOT_COMMIT_REQUEST	
1	HEX	04	LOT_ABORT_REQUEST	
1	HEX	05	LOT_DSNC_COMMAND_REQUEST	
1	HEX	06	LOT_IFI_API_ REQUEST	
1	HEX	07	LOT_SQL_EDF_ REQUEST	
1	HEX	08	LOT_RESYNC_LOST_TO_INITIAL	
1	HEX	09	LOT_SINGLE_PHASE_COMMIT	
1	HEX	0A	LOT_END_OF_TASK_REQUEST	
1	HEX	0B	LOT_IFI_EDF_ REQUEST	
1	HEX	0C	LOT_SPI_REQUEST	
1	HEX	0D	LOT_CICS_SHUTDOWN_REQUEST	
1	HEX	11	LOT_SQL_API_BUT_MUST_ABORT	
1	HEX	16	LOT_IFI_API_BUT_MUST_ABORT	
1	HEX	80	LOT_API_REQUEST_ FAILED	
1	HEX	81	LOT_SQL_API_REQUEST_FAILED	
1	HEX	82	LOT_PREPARE_ ABENDED	
1	HEX	85	LOT_DSNC_COMMAND_REQUEST_FAILED	
1	HEX	86	LOT_IFI_API_REQUEST_FAILED	
Constants for lot_ rmi_return_code				
1	DECIMAL	0	LOT_RMI_RETURN_CODE_OK	
1	HEX	0C	LOT_ABEND_TXN_WITH_DUMP	
Constants for lot_ error_codes				
1	HEX	04	LOT_RCT_TAMPER_ ERROR	
1	HEX	08	LOT_INSTALLATION_ ERROR	
1	HEX	0C	LOT_ATTACH_SHUTDOWN_IN_PROGRESS	
1	HEX	10	LOT_NO_THREAD	
1	HEX	18	LOT_CONN_SUBTASK_ABEND	
1	HEX	1C	LOT_SIGNON_FAILED	
1	HEX	20	LOT_THREAD_RESOURCE_UNAVAILABLE	
1	HEX	24	LOT_CREATE_THREAD_FAILED	
1	HEX	28	LOT_UNKNOWN_CALL	
1	HEX	2C	LOT_RESYNC_FAILED_INITIAL_ START	
1	HEX	34	LOT_ONLY_ DB2_INDOUBT	
1	HEX	38	LOT_CICS_ABORT_DB2_COMMIT	
1	HEX	3C	LOT_DB2_RESOLVE_INDOUBT_ABEND	
1	HEX	40	LOT_ROLLBACK_TXN_FOR_DEADLOCK	
1	HEX	44	LOT_UNKNOWN_ RMI_CALL	
1	HEX	4C	LOT_EDF_CALL_ FAILED	
1	HEX	50	LOT_SHUTDOWN_WHILE_COMMIT_ABORT	
1	HEX	54	LOT_MUST_ABORT	
1	HEX	58	LOT_SINGLE_PHASE_BACKED_OUT	
1	HEX	60	LOT_SINGLE_PHASE_COMMIT_FAILED	
1	HEX	68	LOT_ATTACH_IN_STANDBY_MODE	
1	HEX	70	LOT_ACQUIRE_LOCK_FAILED	
1	HEX	74	LOT_RELEASE_LOCK_FAILED	
1	HEX	78	LOT_AUTH_ TYPE_INVALID	
1	HEX	7C	LOT_RECOVERY_ROUTINE_ENTERED	

Len	Type	Value	Name	Description
1	HEX	80	LOT_INVALID_DDLO_REASON	
1	HEX	84	LOT_INVALID_DDLO_RESPONSE	
1	HEX	88	LOT_INVALID_THREAD_STATE	
1	HEX	8C	LOT_LOST_ OUR_THREAD	
1	HEX	90	LOT_WAIT_ MVS_FAILED	
1	HEX	94	LOT_GETMAIN_FAILED	
1	HEX	98	LOT_INVALID_ RMI_VERB	
1	HEX	9C	LOT_DB2ENTRY_ DISABLED	
1	HEX	A0	LOT_ATTACH_SUBTASK_NO_STORAGE	
1	HEX	A4	LOT_ATTACH_SUBTASK_FAILED	

D2SS Cics/db2 static storage

CONTROL BLOCK NAME = DFHD2SS
DESCRIPTIVE NAME = CICS DB2 attach Static Storage
FUNCTION =
The DFHD2SS block contains global data for the CICS-DB2 connection established during CICS startup before the DFHD2GLB is created. It is also used to store data that needs to survive even if the DB2CONN is discarded and hence the DFHD2GLB freemained.
LIFETIME =
DFHD2SS is getmained by DFHSIB1C during CICS initialisation. Its lifetime is the lifteime of CICS, it is not freemained.
LOCATION =
DFHD2SS resides above the 16MB line. It is anchored off the static storage address list DFHSSAPS which is turn is anchored off the CSA optional features list.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = none
MODULE TYPE = Control block definition
DFHD2SS block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	116	DFHD2SS	
(0)	CHARACTER	16	D2S_PREFIX	Standard Prefix
(0)	HALFWORD	2	D2S_LENGTH	Length of control block
(2)	CHARACTER	14	D2S_EYE	Eyecatcher >DFHD2SS
Anchor addresses				
(10)	ADDRESS	4	D2S_DFHD2GLB	Anchor address of DFHD2GLB
Directory manager tokens for DFHD2ENT and DFHD2TRN control blocks.				
(14)	ADDRESS	4	D2S_D2ENT_ DIR_TOKEN	D2ENT directory token
(18)	ADDRESS	4	D2S_D2TRN_N_DIR_TOKEN	D2TRN dir token (key=name)
(1C)	ADDRESS	4	D2S_D2TRN_T_DIR_TOKEN	D2TRN dir token (key=tranid)
Directory manager token for CSUB. Accessing CSUBs via directory manager is only used in dump formatting				
(20)	ADDRESS	4	D2S_D2CSB_ DIR_TOKEN	D2CSB dir token (key=stck)
Lock manager tokens for locks on the DFHD2GLB, DFHD2ENT and DFHD2TRN control blocks.				
(24)	ADDRESS	4	D2S_D2GLB_ LOCK_TOKEN	DB2CONN lock token
(28)	ADDRESS	4	D2S_D2ENT_ LOCK_TOKEN	D2ENT directory token
(2C)	ADDRESS	4	D2S_D2TRN_ LOCK_TOKEN	D2ENT directory token
Lock manager tokens for locks on CSUB control blocks and LOT control blocks when manipulating double linked chains.				
(30)	ADDRESS	4	D2S_FREE_TCB_LOCK_TOKEN	

Offset Hex	Type	Len	Name (Dim)	Description
				Lock for CSUB free TCBs chns
(34)	ADDRESS	4	D2S_PTHREAD_LOCK_TOKEN	Lock for CSUB prot threads
(38)	ADDRESS	4	D2S_ATHREAD_LOCK_TOKEN	Lock for CSUB active threads
(3C)	ADDRESS	4	D2S_LOT_LOCK_TOKEN	Lock for GWA and RCT LOT chns
Storage manager subpool tokens identifying the subpools for the DFHD2ENT, DFHD2TRN and DFHD2CSB control blocks				
(40)	CHARACTER	8	D2S_D2ENT_SM_TOKEN	D2ENT subpool token
(48)	CHARACTER	8	D2S_D2TRN_SM_TOKEN	D2TRN subpool token
(50)	CHARACTER	8	D2S_D2CSB_SM_TOKEN	D2CSB subpool token
Entry point addresses for CICS-DB2 modules loaded by DFHD2RP				
(58)	ADDRESS	4	D2S_DFHD2CC_ENTRY_POINT	CICS-DB2 Connection Control
(5C)	ADDRESS	4	D2S_DFHD2STR_ENTRY_POINT	CICS-DB2 Start Program
(60)	ADDRESS	4	D2S_DFHD2STP_ENTRY_POINT	CICS-DB2 Stop Program
(64)	ADDRESS	4	D2S_DFHD2TM_ENTRY_POINT	CICS-DB2 Table manager
Counts used to valid DB2ENTRY and DB2TRAN tokens				
(68)	UNSIGNED	4	D2S_DB2ENTRY_CHANGE_COUNT	Count to invalid tokens
(6C)	UNSIGNED	4	D2S_DB2TRAN_CHANGE_COUNT	Count to invalid tokens
Miscellaneous				
(70)	BITSTRING	1	D2S_INIT_ECB	CICS/DB2 initialisation ecb
	1...		*	
	.1..		D2S_INIT_ECB_POSTED	Posted setting for ECB
	..11 1111		*	
(71)	BITSTRING	1	D2S_DISCONNECT_ECB	ECB for disconnecting
(72)	UNSIGNED	1	D2S_D2ST_RESP	Response from restart task
(73)	BITSTRING	1	*	Reserved

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	D2S_D2ST_OK	
1	DECIMAL	2	D2S_D2ST_EXCEPTION	
1	DECIMAL	3	D2S_D2ST_DISASTER	

D2TRN Db2tran block

```
CONTROL BLOCK NAME = DFHD2TRN
DESCRIPTIVE NAME = CICS DB2 attach DB2TRAN control block
FUNCTION =
    The DFHD2TRN block represents a DB2TRAN RDO object, the
    mapping between a DB2ENTRY and a transaction id (transid)
    that is associated with it.
LIFETIME =
    A DFHD2TRN is getmained when a DB2TRAN entity is installed.
    It is freemained when a DB2TRAN is discarded.
LOCATION =
    DFHD2ENT resides above the 16MB line. It is located using
    Directory manager domain using its name as the key. There
    is also a second index using Directory manager so that a
    DFHD2ENT block can be located using the transid it holds.
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS = none
 MODULE TYPE = Control block definition
  DFHD2TRN block
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	44	DFHD2TRN	
(0)	CHARACTER	16	TRN_PREFIX	Standard Prefix
(0)	HALFWORD	2	TRN_LENGTH	
(2)	CHARACTER	14	TRN_EYE	>DFHD2ENT
(10)	CHARACTER	8	TRN_NAME	name of DB2TRAN
(18)	CHARACTER	4	TRN_TRANSID	Transid
(1C)	CHARACTER	8	TRN_DB2ENTRY_ NAME	name of associated DB2ENTRY
(24)	CHARACTER	8	TRN_DB2ENTRY_ ETOKEN	
(24)	ADDRESS	4	TRN_DB2ENTRY_ ADDR	Addr(associated DB2ENTRY)
(28)	UNSIGNED	4	TRN_DB2ENTRY_ COUNT	Count to validate token

Constants

Len	Type	Value	Name	Description
14	CHARACTER	>DFHD2TRN	DFHD2TRN_ EYECATCHER	

FBWAC File browse work area for data tables

```
CONTROL BLOCK NAME = DFHFBWAC
DESCRIPTIVE NAME = CICS (FC) File Browse Work Area
FUNCTION =
     Browse work area for browsing data tables.
     This control block is part of data tables support within
     CICS file control. It is used to keep track of the
     status of a browse to a data table. It is used for both
     shared data tables support and coupling facility data table
     support, although not all fields are used by both.
     An instance of the FBWA represents a browse thread by a
     unit of work to a data table, so there will be one FBWA
     per data table being browsed per UOW that is browsing.
LIFETIME =
     An FBWA is created when a START_BROWSE is issued to a
     data table, and destroyed when the browse is ended.
STORAGE CLASS =
     FBWAs are getmained from one of the FC buffer pools in the
     FC_ABOVE subpool, which is above the line, CICS key stg.
     It is freed back to the buffer pool when the browse ends.
LOCATION =
     The FBWA for a request is addressed by FRT_FBWA_ADDRESS
     in the FRTE.
INNER CONTROL BLOCKS =
     None.
NOTES :
 DEPENDENCIES = S/390
 RESTRICTIONS = None
 MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
      No referenced items are defined outside this control block
  DATA AREAS =
      No fields in operating system data areas are referenced
  CONTROL BLOCKS =
      FBWA_FREE_CHAIN addresses the home buffer chain
  GLOBAL VARIABLES (Macro pass) =
      No global macro variables are referenced
 File Browse Work Area
 This area is used to record status information about a browse
 sequence. It is addressed via a pointer in the FRTE associated
 with the browse and created using an IO buffer of appropriate
 size obtained from a file control IO buffer pool.
 Some of the data relate to the state of the browse as perceived
 at the API, e.g. whether the browse is GENERIC and what key was
 last returned to the application.
 CMT-specific fields
 Because browsing a CICS-maintained shared data table may
 require references to the source data set it may be necessary
 institute a source browse. Some data in the FBWA relate to the
 state of any such browse and its relationship to the API browse.
```

The following is an explanation of some of the less immediately obvious items which refer to the source data set browse.

FBWA_SOURCE_CURRENT

is meaningful only if FBWA_SOURCE_STARTED is on. It shows that the last browse request was satisfied by reference to the source so the next one could validly be processed by simply passing the request on to the source browse service.

FBWA_SOURCE_IN_SEQ

is meaningful only if FBWA_SOURCE_STARTED is on. It shows that the browse is full key GTEQ and that the source browse is known to be positioned at a key less than or equal to that of the current API browse position.
It is used to determine whether a RESETBR can be safely omitted in some cases where recourse to the source browse is necessary to satisfy a request.
It is used solely for optimization and is set only in circumstances in which it is easy to be sure of its truth.
SOURCE_IN_SEQ is used to hold the value of the flag at the start of a request and the flag itself is set off. It is set on again at the end of the request if appropriate.

FBWA_TOKEN_VALID

shows that the last browse request was satisfied from the table and that the token in the FRTE, FRT_DT_RECORD_TOKEN, corresponds to the current browse key FBWA_CURRENT_KEY.
The token is used to optimize table access for sequential browse requests by avoiding the index search.
This field is also used for UMTs.

FBWA_NEXT_KEY_VALID

shows that the key in FBWA_NEXT_KEY is valid. If a gap is encountered while browsing a table SDTF returns the next key in the table. This is copied into FBWA_NEXT_KEY and FBWA_NEXT_KEY_VALID is set on. As long as the browse remains sequential, no attempt will be made to revert to table retrieval until this key value is reached.

FBWA_SEQUENTIAL

shows that the next browse request may be treated as sequential provided that it satisfies the criteria. The indicator is set only after a request has completed with an OK or ENDFILE response so that continuation in any other case, e.g. after NOTFND, will be treated as a reposition.
This field is also used for UMTs and CFDTs.

UMT-specific fields
There are no fields used exclusively for UMTs.

CFDT-specific fields
There are no fields used exclusively for CFDTs.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	48	FBWA	
(0)	CHARACTER	48	FBWA_FIXED_PART	Fixed format part of FBWA
(0)	CHARACTER	16	FBWA_EYE_ CATCHER	Eye catcher
(0)	HALFWORD	2	FBWA_LENGTH	Length of used part
(2)	CHARACTER	6	FBWA_EYE1	>DFHFC
(8)	CHARACTER	8	FBWA_EYE2	FBWA
(10)	BITSTRING	1	FBWA_FLAGS1	Type of request indicators
	1...		FBWA_RBA	Browsing by RBA
	.1..		FBWA_BACKWARDS	Browsing backwards
	..1.		FBWA_GTEQ	Browse is GTEQ
	...1		FBWA_GENERIC	Browse is GENERIC
	.... 1...		FBWA_FIRST	Last request was STARTBR or RESETBR
	.... .111		*	Reserved
(11)	BITSTRING	1	FBWA_FLAGS2	More indicators
	1...		FBWA_TOKEN_ VALID	Table token corresponds to current key
	.1..		FBWA_SOURCE_STARTED	Source browse initiated
	..1.		FBWA_SOURCE_CURRENT	Source browse is correctly positioned
	...1		FBWA_SOURCE_ IN_SEQ	Source browse is FKGE and not later than current key
	.... 1...		FBWA_NEXT_ KEY_VALID	End of gap key is valid
	.... .1..		FBWA_SEQUENTIAL	Sequential is allowed for next browse request
	11		*	Reserved
(12)	HALFWORD	2	FBWA_KEY_LENGTH	Current browse key length
(14)	ADDRESS	4	FBWA_FREE_CHAIN	Home buffer chain
(18)	ADDRESS	4	FBWA_CURRENT_ KEY	Current key field address
(1C)	ADDRESS	4	FBWA_REQUEST_ KEY	Request key field address
(20)	ADDRESS	4	FBWA_NEXT_KEY	End of gap key address
(24)	CHARACTER	12	FBWA_RECORD_ TOKEN	Current key table token
(30)	CHARACTER		FBWA_FIXED_END	End of fixed part
(30)	CHARACTER		FBWA_KEYS	Start of key fields

FCPEC File control cfdt pool element

CONTROL BLOCK NAME = DFHFCPEC
DESCRIPTIVE NAME = CICS FC Pool Element (FCPE)
FUNCTION =

DFHFCPE describes the DSECT for a File Control Pool Element. A pool element represents one connection to a Coupling Facility Data Table Pool.
Coupling Facility Data Tables are organised into pools, each of which is similar in scope and function to a CICS FOR.
For each table pool which can be accessed by a given MVS image, there is a table server region running in that image which manages access to the pool.
A pool element is created and chained to FC static when a file definition that refers to the pool is installed and there is not already a pool element for that CFDR pool.
A connection to the CFDT server is made when CICS opens the first table for the pool, and a flag in the FCPE is set to indicate that the pool is now connected.
If the CFDT server goes down the FCPE will be marked connect_failed when CICS realises the server has gone. This flag is only reset when the server returns and a new connection is successfully made. Note : it is important that the testing of the connect_failed flag is always serialised with any connect that may already be in progress, by waiting on the connect complete ECB.
The address of the head of the FCPE chain in FC Static is field FC_FCPE_CHAIN.
FCPEs are getmained from the FCPE subpool which is created by DFHFCRP during File Control Initialisation.
File Control Pool Elements are freemained by DFHFCSD at CICS shutdown when pool disconnections are issued.

LIFETIME =

Created during installation of a file definition that refers to the associated pool.
Deleted at shutdown (when disconnects are also issued for all pools to which CICS is currently connected).

STORAGE CLASS =

Above 16M line. CICS key.

LOCATION =
INNER CONTROL BLOCKS = None.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Control block definition

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	76	DFHFCPE	
Eye catcher				
(0)	CHARACTER	16	FCPE_EYE_CATCHER	Eye catcher
(0)	UNSIGNED	2	FCPE_LENGTH	Length of FCPE
(2)	CHARACTER	6	FCPE_EYE1	>DFHFC FC 'domain'
(8)	CHARACTER	8	FCPE_EYE2	FCPE
Main part of FCPE				
(10)	CHARACTER	60	FCPE_MAIN_PART	Main part of FCPE
(10)	ADDRESS	4	FCPE_NEXT_ ADDRESS	next in chain
(14)	ADDRESS	4	FCPE_PREV_ ADDRESS	prev in chain
(18)	CHARACTER	8	FCPE_POOL_NAME	name of pool
(20)	ADDRESS	4	FCPE_CONNECTION_ TOKEN	connection token
(24)	FULLWORD	4	FCPE_COUNT_ OF_OPENS	CFDTs open for pool
(28)	FULLWORD	4	FCPE_INSTANCE_ NUMBER	server instance
(2C)	BITSTRING	1	FCPE_FLAGS	flags
	1...		FCPE_CONNECT_ FAILED	server failed
	.1..		FCPE_RESTARTED	a restart call to the server has been issued successfully
	..1.		FCPE_CONNECT_ IN_PROGRESS	a CONNECT to this pool is in progress
	...1 1111		*	reserved
(2D)	CHARACTER	3	*	reserved

Offset Hex	Type	Len	Name (Dim)	Description
(30)	ADDRESS	4	FCPE_LOCK_TOKEN	Lock token used for serialisation
(34)	FULLWORD	4	FCPE_LRS_COUNT	Number of free locking request slots (LRSs)
(38)	CHARACTER	8	FCPE_LRS_ WAIT_HEAD	Chain head for chain of LRS waiters
(38)	ADDRESS	4	FCPE_FIRST_ LRS_WAITER	first LRS waiter in chain
(3C)	ADDRESS	4	FCPE_LAST_ LRS_WAITER	last LRS waiter in chain
(40)	CHARACTER	8	FCPE_WAIT_HEAD	Chain head for chain of maxreqs waiters
(40)	ADDRESS	4	FCPE_FIRST_ WAITER	first maxreqs waiter in chain
(44)	ADDRESS	4	FCPE_LAST_ WAITER	last maxreqs waiter in chain
(48)	ADDRESS	4	FCPE_OPEN_ FILE_CHAIN	anchor for chain of files open against CFDTs in pool

FCPWC File control cfdt pool wait element

CONTROL BLOCK NAME = DFHFCPWC
DESCRIPTIVE NAME = CICS FC CFDT Pool Wait Element
FUNCTION =

DFHFCPW describes the DSECT for a File Control CFDT Pool Wait Element. A pool wait element represents a task which has tried to issue a request to a coupling facility data table that resides in a particular server pool, but which has to wait because the number of requests allowed in the server at any one time has been reached. Depending on the kind of request, the FCPW will represent either a 'Locking request slot' waiter or a 'MaxReqs' waiter.

A Locking request slot waiter is a Locking request (one which will acquire locks) that has to wait because all the slots allocated to Locking requests are currently in use.

A MaxReqs waiter is a non-locking request which has to wait because the maximum number of requests (of any kind) allowed in the server has been exceeded. Thus the Locking request slots are a subset of the MaxReqs slots.

Different kinds of waiter are chained on separate queues.

When a request has to wait, it needs to be appended to a chain anchored from the pool element, and unchained when the request can be resumed. The different kinds of waiter are chained on separate wait queues.

FCPWs are getmained from the FCPW subpool which is created by DFHFCRP during File Control Initialisation.

A file control CFDT Pool Wait Element is freemained when the waiter that it represents has been successfully resumed.

The FCPW contains the following fields:

- Pointer to next FCPW in chain
- Pointer to previous FCPW in chain
- Suspend token
- Task token for the waiting task
- Suspend start time (for monitoring)
- Transaction number (for debug - so it appears in a dump)
- The priority at which the task should be resumed (it will be set to a higher priority when it is dequeued, to give it more chance of restarting)
- Some flags, indicating: type of waiter

LIFETIME =

The lifetime of an FCPW is the time during which the waiter task has to wait. It is created by the module issuing the request when it is discovered that the request will have to wait, and destroyed by that module when the request is resumed.

STORAGE CLASS =

Above 16M line. CICS key.

LOCATION =

The addresses for the heads of the different FCPW wait chains are in the pool element for the server pool being accessed, in fields FCPE_LRS_WAIT_CHAIN (for the Locking request slot waiters) and FCPE_WAIT_CHAIN (for the MaxReqs waiters).

INNER CONTROL BLOCKS = None
NOTES :
DEPENDENCIES = S/390
RESTRICTIONS = None
MODULE TYPE = Control block definition
EXTERNAL REFERENCES = None
DATA AREAS = None
CONTROL BLOCKS = None
GLOBAL VARIABLES (Macro pass) = None

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	48	DFHFCPW	
Eye catcher for FC CFDT Pool Wait element				
(0)	CHARACTER	16	FCPW_EYE_CATCHER	Eye catcher
(0)	UNSIGNED	2	FCPW_LENGTH	Length of FCPW
(2)	CHARACTER	6	FCPW_EYE1	>DFHFC FC 'domain'
(8)	CHARACTER	8	FCPW_EYE2	FCPW
Main part of FC CFDT Pool Wait element				
(10)	CHARACTER	32	FCPW_MAIN_PART	Main part of FCPW
(10)	CHARACTER	8	FCPW_CHAIN	chaining fields
(10)	ADDRESS	4	FCPW_NEXT_ ADDRESS	next in chain
(14)	ADDRESS	4	FCPW_PREV_ ADDRESS	prev in chain
(18)	ADDRESS	4	FCPW_SUSPEND_ TOKEN	suspend token
(1C)	ADDRESS	4	FCPW_TASK_TOKEN	Task token for waiting task
(20)	CHARACTER	8	FCPW_SUSPEND_ TIME	suspend time (for monitoring)

Offset Hex	Type	Len	Name (Dim)	Description
(28)	UNSIGNED	1	FCPW_RESUME_ PRIORITY	priority at which task should be resumed
(29)	BITSTRING	1	FCPW_FLAGS	flags
	1...		FCPW_LRS_WAIT	wait is for a Locking request slot
	.1..		FCPW_MAXREQS_ WAIT	wait is for a MaxReqs slot
	..11 1111		*	reserved
(2A)	CHARACTER	2	*	reserved
(2C)	FULLWORD	4	FCPW_TRAN_NUM	Transaction number (for debug purposes)

FCQRE File control quiesce receive element

-

File Control Quiesce Receive Element

Declare the FC Quiesce Receive Element (FCQRE) and associated structures and constants.

-

Element

Each quiesce request received from VSAM RLS via the quiesce exit results in DFHFCQX, the quiesce exit module, creating an FCQRE which is passed to DFHFCQR, the quiesce receive system task module. FCQREs reside in MVS getmained storage because DFHFCQX has no access to CICS services. They are chained in a one-way linked list anchored in FC static field FC_FCQRE_FIRST.

Because DFHFCQX runs under a different MVS TCB to DFHFCQR, standard compare-and-swap chain manipulation logic is used when processing the chain. DFHFCQX adds a new FCQRE to the front of the chain. DFHFCQR isolates the chain then reverses the order of the FCQREs so that processing occurs oldest first. The isolated chain is anchored in FC static field FC_FCQRE_ISOLATE.

There is also a permanent Error FCQRE used for communicating errors between DFHFCQX and DFHFCQR. This is addressed from FC static field FC_FCQRE_ERROR, and is added to the chain when an error occurs.

All FCQREs appear in a CICS system dump, including the Error FCQRE if it is in use at the time.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	96	DFHFCQRE	
(0)	CHARACTER	24	FCQRE_PREFIX	
(0)	HALFWORD	2	FCQRE_LENGTH	length
(2)	CHARACTER	1	FCQRE_ARROW	'>'
(3)	CHARACTER	3	FCQRE_DFH	'DFH'
(6)	CHARACTER	2	FCQRE_DOMAIN	'FC'
(8)	CHARACTER	8	FCQRE_BLOCKNAME	'QRE'
(10)	ADDRESS	4	FCQRE_NEXT	-> next new fcqre
(14)	ADDRESS	4	FCQRE_NEXT_ ISOLATE	-> next isolated fcqre
(18)	CHARACTER	72	FCQRE_BODY	
(18)	CHARACTER	44	FCQRE_DATASET	dataset name
(18)	CHARACTER	16	FCQRE_CACHE	cache name
(44)	UNSIGNED	1	FCQRE_ELEMENT_ TYPE	type of element
(45)	UNSIGNED	1	FCQRE_QUIESCE_ TYPE	type of quiesce request
(46)	UNSIGNED	1	FCQRE_ERROR_ TYPE	type of error request
(47)	BITSTRING	1	FCQRE_FLAGS	flags
	1...		FCQRE_IMMEDIATE	1=immediate close
	.1..		FCQRE_CONCURRENT	1=concurrent copy technique
	..1.		FCQRE_ERROR_ USED	1=error fcqre & in use
	...1 1111		*	reserved
(48)	CHARACTER	8	FCQRE_QUICMP_ TOKEN	token to return to vsam rls on quicmp call
(50)	UNSIGNED	4	FCQRE_ERROR_ DATA	error data if error request

Offset Hex	Type	Len	Name (Dim)	Description
(54)	UNSIGNED	4	FCQRE_DATASET_ LENGTH	sig length dataset name@P1C
(54)	UNSIGNED	4	FCQRE_CACHE_ LENGTH	sig length cache name
(58)	CHARACTER	8	*	reserved

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	FCQRE_QUIESCE_ REQUEST	
1	DECIMAL	2	FCQRE_ERROR_ REQUEST	
1	DECIMAL	1	FCQRE_QUIESCE	quiclose
1	DECIMAL	2	FCQRE_UNQUIESCE	quiopen
1	DECIMAL	3	FCQRE_NONBWO_START	quicopy
1	DECIMAL	4	FCQRE_NONBWO_END	quicend
1	DECIMAL	5	FCQRE_BWO_START	quibwo
1	DECIMAL	6	FCQRE_BWO_END	quibend
1	DECIMAL	7	FCQRE_LOCKS_ RECOV_COMPLETE	quillrc
1	DECIMAL	8	FCQRE_FWD_ RECOV_COMPLETE	quifrc
1	DECIMAL	9	FCQRE_CACHE_ AVAILABLE	quica
1	DECIMAL	1	FCQRE_STG_FAILURE	storage obtain macro failed in quiesce exit
8	CHARACTER	QRE	FCQRE_EYE	eyecatcher

FCQSE File control quiesce send element

-

File Control Quiesce Send Element

Declare the FC Quiesce Send Element (FCQSE) and associated structures and constants.

-

Element

Each quiesce request initiated by CICS results in DFHFCQI, the quiesce initiate module, creating an FCQSE which is passed to DFHFCQS, the quiesce send module. FCQSEs reside in subpool FC_ABOVE, the token for which is in FC static. They are chained in a two-way linked list anchored in FC static fields FC_FCQSE_FIRST and FC_FCQSE_LAST.

FCQSEs are added to the end of the chain by DFHFCQI. The chain is scanned from the front by DFHFCQS, so the oldest FCQSE is processed first.

All FCQSEs appear in a CICS system dump.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	112	DFHFCQSE	
(0)	CHARACTER	24	FCQSE_PREFIX	
(0)	HALFWORD	2	FCQSE_LENGTH	length
(2)	CHARACTER	1	FCQSE_ARROW	'>'
(3)	CHARACTER	3	FCQSE_DFH	'DFH'
(6)	CHARACTER	2	FCQSE_DOMAIN	'FC'
(8)	CHARACTER	8	FCQSE_BLOCKNAME	'QSE'
(10)	ADDRESS	4	FCQSE_NEXT	-> next fcqse
(14)	ADDRESS	4	FCQSE_PREV	-> prev fcqse
(18)	CHARACTER	88	FCQSE_BODY	
(18)	CHARACTER	44	FCQSE_DSNAME	dataset name
(44)	UNSIGNED	1	FCQSE_QUIESCE_ TYPE	type of quiesce request
(45)	BITSTRING	1	FCQSE_FLAGS	flags
	1...		FCQSE_WAIT	1=wait for completion
	.1..		FCQSE_CICS	1=cics initiated

Offset Hex	Type	Len	Name (Dim)	Description
	..11 1111		*	reserved
(46)	UNSIGNED	1	FCQSE_RESP_CODE	response from request
(47)	UNSIGNED	1	FCQSE_STATE	element state
(48)	UNSIGNED	4	FCQSE_SUSPEND_ TOKEN	suspend/resume token
(4C)	ADDRESS	4	FCQSE_VSAM_ ECB_ADDR	-> vsam rls ecb
(50)	UNSIGNED	4	FCQSE_TIMEOUT_ TIME	timeout time (secs)
(54)	UNSIGNED	1	FCQSE_CONFLICT	type of conflicting quiesce
(55)	CHARACTER	3	*	reserved
(58)	CHARACTER	10	FCQSE_USERID	userid of initiating task
(62)	CHARACTER	2	FCQSE_VSAM_RC	vsam rls codes
(62)	UNSIGNED	1	FCQSE_R15	gpr 15
(63)	UNSIGNED	1	FCQSE_REASON	reason code
(64)	CHARACTER	4	FCQSE_TRAN_ NUMBER	xm transaction number of initiating task
(68)	FULLWORD	4	FCQSE_DSNAME_ LENGTH	sig length of dsname
(6C)	CHARACTER	4	*	reserved

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	STCK_TYPE	store clock data type
(0)	UNSIGNED	4	APPROX_SECONDS	top word approxes to secs
(4)	UNSIGNED	4	REST_OF_STCK	rest of store clock

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	FCQSE_QUIESCE	quiesce
1	DECIMAL	2	FCQSE_IMMQUIESCE	immed quiesce
1	DECIMAL	3	FCQSE_UNQUIESCE	unquiesce
1	DECIMAL	4	FCQSE_NONBWO_ CANCEL	cancel of a non-bwo backup
1	DECIMAL	5	FCQSE_BWO_CANCEL	cancel of a bwo backup
1	DECIMAL	6	FCQSE_QUIESCE_ CANCEL	cancel of a quiesce
1	DECIMAL	1	FCQSE_OK	successful
1	DECIMAL	3	FCQSE_UNKNOWN_ VSAM_DATASET	unknown
1	DECIMAL	4	FCQSE_QUIESCE_ NOT_POSSIBLE	conflict
1	DECIMAL	5	FCQSE_UNQUIESCE_ NOT_POSSIBLE	conflict
1	DECIMAL	7	FCQSE_CANCELLED	cancelled
1	DECIMAL	8	FCQSE_TIMED_OUT	timedout
1	DECIMAL	9	FCQSE_IOERR	i/o error
1	DECIMAL	10	FCQSE_SERVER_ FAILURE	no server
1	DECIMAL	11	FCQSE_DATASET_ MIGRATED	migrated
1	DECIMAL	12	FCQSE_VSAM_ERROR	sms abend
1	DECIMAL	13	FCQSE_USER_ NOT_AUTH	not auth
1	DECIMAL	1	FCQSE_NEW_STATE	
1	DECIMAL	2	FCQSE_SENT_STATE	
1	DECIMAL	3	FCQSE_TIMEDOUT_ STATE	
1	DECIMAL	4	FCQSE_RESUMED_ STATE	
1	DECIMAL	1	FCQSE_CONF_QUIESCE	quiesce
1	DECIMAL	2	FCQSE_CONF_ UNQUIESCE	unquiesce
1	DECIMAL	3	FCQSE_CONF_NONBWO	non-bwo backup
1	DECIMAL	4	FCQSE_CONF_BWO	bwo backup
1	DECIMAL	5	FCQSE_CONF_UNKNOWN	unknown
8	CHARACTER	QSE	FCQSE_EYE	eyecatcher

FCUPC File control cfdt uow pool block

```
CONTROL BLOCK NAME = DFHFCUPC
DESCRIPTIVE NAME = CICS (FC) CFDT UOW Pool Block
FUNCTION =
    The FCUP block represents recoverable updates made within
    a unit of work to tables within a coupling facility data
    table pool.
    THE FCUP block is used by the CF data tables part of the
    File Control component. Each FCUP block represents the
    RMC link to a CF data table pool within a unit of work.
    This means that within a unit of work, each CF data table
    pool which contains one or more CF data tables to which
    the UOW has made recoverable updates will be represented
    by an FCUP block: there is one FCUP block per UOW per
    recoverably-updated CFDT pool.
    FCUP blocks are getmained from the FCUP subpool which is
    created by DFHFCRP during File Control Initialisation.
LIFETIME =
    The lifetime of an FCUP block is the same as that of the
    RMC Link which it represents.
    An FCUP block is created by the CF data tables request
    processor, DFHFCDR, when the first recoverable update is
    made within a unit of work to a table which resides in the
    CF data table pool to which the FCUP block will refer.
    The FCUP block is created at the same time as an RMC link
    is created, and it represents File Control's interest in
    that link.
    The FCUP block is freed at syncpoint time by the CFDT
    Syncpoint processor, DFHFCDW, at the successful completion
    of syncpoint for that pool within the unit of work.
STORAGE CLASS =
    Above 16M line. CICS key.
LOCATION =
    The FCUP blocks for a unit of work are chained from the
    FRAB, addressed by FRAB_FCUP_CHAIN_ADDRESS.
INNER CONTROL BLOCKS =
    None
NOTES :
 DEPENDENCIES = S/390
 RESTRICTIONS = None
 MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
     None
  DATA AREAS =
     None
  CONTROL BLOCKS =
     THE FCUP block contains pointer to the pool element for
     the CFDT pool it represents, and a back-pointer to the
     FRAB from which it is chained.
  GLOBAL VARIABLES (Macro pass) =
     None
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	44	DFHFCUP	
	Eye catcher for FC CFDT UOW Pool Block			
(0)	CHARACTER	16	FCUP_EYE_CATCHER	Eye catcher
(0)	UNSIGNED	2	FCUP_LENGTH	Length of FCUP
(2)	CHARACTER	6	FCUP_EYE1	>DFHFC FC 'domain'
(8)	CHARACTER	8	FCUP_EYE2	FCUP
	Main part of FC CFDT UOW Pool Block			
(10)	CHARACTER	28	FCUP_MAIN_PART	Main part of FCUP
(10)	CHARACTER	8	FCUP_CHAIN	chaining fields
(10)	ADDRESS	4	FCUP_NEXT_ ADDRESS	next in chain
(14)	ADDRESS	4	FCUP_PREV_ ADDRESS	prev in chain
(18)	CHARACTER	8	FCUP_POOL_NAME	CFDT Pool Name
(20)	ADDRESS	4	FCUP_LINK_TOK	RMC Link Token
(24)	ADDRESS	4	FCUP_POOL_ ELEM_PTR	Pointer to FCPE
(28)	ADDRESS	4	FCUP_FRAB_PTR	Back-pointer to FRAB

FEP01 Frontend programming interface trace

Offset Hex	Type	Len	Name (Dim)	Description
2	HEX	1200	SZ_TRP_API_ENTRY	
2	HEX	1201	SZ_TRP_API_EXIT	
2	HEX	1220	SZ_TRP_SPI_ENTRY	
2	HEX	1221	SZ_TRP_SPI_EXIT	

```
  ================================================================
= =
  = X'1240' -> X'125F' are for the FEPI Resource Manager =
  = SZ3000 -> SZ3999 Adapter program usage =
= =
  ================================================================
```

Offset Hex	Type	Len	Name (Dim)	Description
2	HEX	1240	SZ_TRP_ADA_ENTRY	
2	HEX	1241	SZ_TRP_ADA_EXIT	
2	HEX	1242	SZ_TRP_ADA_CHECK	
2	HEX	1243	SZ_TRP_ADA_BRM	
2	HEX	1244	SZ_TRP_ADA_ARM	
2	HEX	1245	SZ_TRP_ADA_BXB	
2	HEX	1246	SZ_TRP_ADA_AXB	
2	HEX	1247	SZ_TRP_ADA_BXA	
2	HEX	1248	SZ_TRP_ADA_AXA	
2	HEX	1250	SZ_TRP_ADA_ GET_FAIL	
2	HEX	1251	SZ_TRP_ADA_ WAIT_FAIL	

```
  ================================================================
= =
  = X'1260' -> X'12BF' are for the FEPI Resource Manager =
  = SZ4000 -> SZ5999 usage =
= =
  ================================================================
```

Offset Hex	Type	Len	Name (Dim)	Description
2	HEX	1260	SZ_TRP_SIP_ENTRY	
2	HEX	1261	SZ_TRP_SIP_EXIT	
2	HEX	1262	SZ_TRP_SIP_ERR_SIT	
2	HEX	1263	SZ_TRP_SIP_ ERR_STATE	
2	HEX	1264	SZ_TRP_SIP_ERR_ENQ	
2	HEX	1265	SZ_TRP_SIP_ERR_SP	
2	HEX	1266	SZ_TRP_SIP_ ERR_RUNAWAY	
2	HEX	1267	SZ_TRP_SIP_ERR_CHP	
2	HEX	1268	SZ_TRP_SIP_ ERR_SWOP	
2	HEX	1269	SZ_TRP_SIP_REENTER	
2	HEX	126A	SZ_TRP_SIP_ABEND	
2	HEX	126B	SZ_TRP_ZNG_ENTRY	
2	HEX	126C	SZ_TRP_ZNG_EXIT	
2	HEX	126D	SZ_TRP_ZNG_ GET_GOOD	
2	HEX	126E	SZ_TRP_ZNG_ GET_FAIL	
2	HEX	126F	SZ_TRP_ZAG_ENTRY	
2	HEX	1270	SZ_TRP_ZAG_EXIT	
2	HEX	1271	SZ_TRP_ZAG_ GET_GOOD	
2	HEX	1272	SZ_TRP_ZAG_ GET_FAIL	
2	HEX	1273	SZ_TRP_ZRG_ENTRY	
2	HEX	1274	SZ_TRP_ZRG_EXIT	
2	HEX	1275	SZ_TRP_ZRG_ GET_GOOD	
2	HEX	1276	SZ_TRP_ZRG_ GET_FAIL	
2	HEX	1277	SZ_TRP_ZFR_ENTRY	
2	HEX	1278	SZ_TRP_ZFR_ FREE1_GOOD	
2	HEX	1279	SZ_TRP_ZFR_ FREE1_FAIL	
2	HEX	127A	SZ_TRP_ZFR_ FREE2_GOOD	
2	HEX	127B	SZ_TRP_ZFR_ FREE2_FAIL	
2	HEX	127C	SZ_TRP_ZFR_EXIT	

API related trace point allocations 1400 ->

Offset Hex	Type	Len	Name (Dim)	Description
2	HEX	1400	SZ_TRP_RPW_ENTRY	
2	HEX	1401	SZ_TRP_RPW_EXIT	
2	HEX	1402	SZ_TRP_RRT_ENTRY	
2	HEX	1403	SZ_TRP_RRT_ FREE_DQE	
2	HEX	1404	SZ_TRP_RRT_ FREE_DYN	
2	HEX	1405	SZ_TRP_RRT_EXIT	
2	HEX	1406	SZ_TRP_RQW_ENTRY	
2	HEX	1407	SZ_TRP_RQW_QUEUE	
2	HEX	1408	SZ_TRP_RQW_POST	
2	HEX	1409	SZ_TRP_RQW_EXIT	
2	HEX	140A	SZ_TRP_RDP_ENTRY	
2	HEX	140B	SZ_TRP_RDP_ INITDONE	
2	HEX	140C	SZ_TRP_RDP_PROCESS	
2	HEX	140D	SZ_TRP_RDP_BAD_REQ	
2	HEX	140E	SZ_TRP_RDP_POST	
2	HEX	140F	SZ_TRP_RDP_IDLE	
2	HEX	1410	SZ_TRP_RDP_FORCED	
2	HEX	1411	SZ_TRP_RDP_ NO_COMMON	
2	HEX	1412	SZ_TRP_RDP_NO_LIFO	
2	HEX	1413	SZ_TRP_RDP_EXIT	

Offset Hex	Type	Len	Name (Dim)	Description
2	HEX	1414	SZ_TRP_RNO_ENTRY	
2	HEX	1415	SZ_TRP_RNO_EXIT	
2	HEX	1416	SZ_TRP_RII_ENTRY	
2	HEX	1417	SZ_TRP_RII_EXIT	
2	HEX	1418	SZ_TRP_RID_ENTRY	
2	HEX	1419	SZ_TRP_RID_EXIT	
2	HEX	141A	SZ_TRP_RZZ_ENTRY	
2	HEX	141B	SZ_TRP_RZZ_EXIT	
2	HEX	141C	SZ_TRP_RNC_ENTRY	
2	HEX	141D	SZ_TRP_RNC_EXIT	
2	HEX	141E	SZ_TRP_RCA_ENTRY	
2	HEX	141F	SZ_TRP_RCA_FREE	
2	HEX	1420	SZ_TRP_RCA_ CLOSE_ACB	
2	HEX	1421	SZ_TRP_RCA_EXIT	
2	HEX	1422	SZ_TRP_RIO_ENTRY	
2	HEX	1423	SZ_TRP_RIO_ DEFACB_ERROR	
2	HEX	1424	SZ_TRP_RIO_EXIT	
2	HEX	1425	SZ_TRP_RIN_ENTRY	
2	HEX	1426	SZ_TRP_RIN_ERROR	
2	HEX	1427	SZ_TRP_RIN_GETMAIN	
2	HEX	1428	SZ_TRP_RIN_EXIT	
2	HEX	1429	SZ_TRP_RIP_ENTRY	
2	HEX	142A	SZ_TRP_RIP_ERROR	
2	HEX	142B	SZ_TRP_RIP_GETMAIN	
2	HEX	142C	SZ_TRP_RIP_EXIT	
2	HEX	142D	SZ_TRP_RIT_ENTRY	
2	HEX	142E	SZ_TRP_RIT_ERROR	
2	HEX	142F	SZ_TRP_RIT_GETMAIN	
2	HEX	1430	SZ_TRP_RIT_EXIT	
2	HEX	1431	SZ_TRP_RIS_ENTRY	
2	HEX	1432	SZ_TRP_RIS_ERROR	
2	HEX	1433	SZ_TRP_RIS_GETMAIN	
2	HEX	1434	SZ_TRP_RIS_EXIT	
2	HEX	1435	SZ_TRP_RIC_ENTRY	
2	HEX	1436	SZ_TRP_RIC_ERROR	
2	HEX	1437	SZ_TRP_RIC_GETMAIN	
2	HEX	1438	SZ_TRP_RIC_EXIT	
2	HEX	1439	SZ_TRP_RDG_ENTRY	
2	HEX	143A	SZ_TRP_RDG_FREE	
2	HEX	143B	SZ_TRP_RDG_ BAD_POOL	
2	HEX	143C	SZ_TRP_RDG_EXIT	
2	HEX	143D	SZ_TRP_RDC_ENTRY	
2	HEX	143E	SZ_TRP_RDC_EXIT	
2	HEX	143F	SZ_TRP_RDS_ENTRY	
2	HEX	1440	SZ_TRP_RDS_FREE	
2	HEX	1441	SZ_TRP_RDS_ BAD_PROPSET	
2	HEX	1442	SZ_TRP_RDS_EXIT	
2	HEX	1443	SZ_TRP_RDN_ENTRY	
2	HEX	1444	SZ_TRP_RDN_FREE	
2	HEX	1445	SZ_TRP_RDN_ BAD_NODE	
2	HEX	1446	SZ_TRP_RDN_EXIT	
2	HEX	1447	SZ_TRP_RDT_ENTRY	
2	HEX	1448	SZ_TRP_RDT_FREE	
2	HEX	1449	SZ_TRP_RDT_ BAD_TARGET	
2	HEX	144A	SZ_TRP_RDT_EXIT	
2	HEX	144B	SZ_TRP_RSC_ENTRY	
2	HEX	144C	SZ_TRP_RSC_ UNKNOWN_LUTYPE	
2	HEX	144D	SZ_TRP_RSC_EXIT	
2	HEX	144E	SZ_TRP_VQS_ENTRY	
2	HEX	144F	SZ_TRP_VQS_EXIT	
2	HEX	1450	SZ_TRP_RIW_ENTRY	
2	HEX	1451	SZ_TRP_RIW_EXIT	
2	HEX	1452	SZ_TRP_RIF_ENTRY	
2	HEX	1453	SZ_TRP_RIF_EXIT	
2	HEX	1454	SZ_TRP_RIA_ENTRY	
2	HEX	1459	SZ_TRP_RIA_EXIT	
2	HEX	145A	SZ_TRP_RIQ_ENTRY	
2	HEX	145B	SZ_TRP_RIQ_EXIT	
2	HEX	145C	SZ_TRP_RXD_ENTRY	
2	HEX	145D	SZ_TRP_RXD_EXIT	
2	HEX	145E	SZ_TRP_RRD_ENTRY	
2	HEX	145F	SZ_TRP_RRD_EXIT	
2	HEX	1460	SZ_TRP_RSE_ENTRY	
2	HEX	1461	SZ_TRP_RSE_EXIT	
2	HEX	1462	SZ_TRP_RCT_ENTRY	
2	HEX	1463	SZ_TRP_RCT_EXIT	
2	HEX	1464	SZ_TRP_RID_ FREE_DSR	
2	HEX	1465	SZ_TRP_RIO_FREE	
2	HEX	1466	SZ_TRP_RIO_GETMAIN	
2	HEX	1467	SZ_TRP_RDC_FREE	
2	HEX	1468	SZ_TRP_2CP_ENTRY	
2	HEX	1469	SZ_TRP_2CP_EXIT	
2	HEX	146A	SZ_TRP_PCP_ENTRY	

Offset Hex	Type	Len	Name (Dim)	Description
2	HEX	146B	SZ_TRP_PCP_EXIT	
2	HEX	146C	SZ_TRP_VRA_ENTRY	
2	HEX	146D	SZ_TRP_VRA_EXIT	
2	HEX	146E	SZ_TRP_RIO_GETFAIL	
2	HEX	146F	SZ_TRP_RIO_GETLIST	
2	HEX	1470	SZ_TRP_RIO_GENCB_ERROR	
2	HEX	1471	SZ_TRP_RIO_OPENACB_ERROR	
2	HEX	1472	SZ_TRP_RQR_ENTRY	
2	HEX	1473	SZ_TRP_RQR_EXIT	
2	HEX	1474	SZ_TRP_RIC_GETDSR	
2	HEX	1475	SZ_TRP_RIC_GETDCD	
2	HEX	1476	SZ_TRP_2SB_ENTRY	
2	HEX	1477	SZ_TRP_2SB_BEFOREO	
2	HEX	1478	SZ_TRP_2SB_BEFORES	
2	HEX	1479	SZ_TRP_2SB_EXIT	
2	HEX	147A	SZ_TRP_2SC_ENTRY	
2	HEX	147B	SZ_TRP_2SC_EXIT	
2	HEX	1480	SZ_TRP_2SD_ENTRY	
2	HEX	1481	SZ_TRP_2SD_BEFORES	
2	HEX	1482	SZ_TRP_2SD_EXIT	
2	HEX	1483	SZ_TRP_2ID_ENTRY	
2	HEX	1484	SZ_TRP_2ID_BEFORES	
2	HEX	1485	SZ_TRP_2ID_BEFOREP	
2	HEX	1486	SZ_TRP_2ID_EXIT	
2	HEX	1487	SZ_TRP_2OA_ENTRY	
2	HEX	1488	SZ_TRP_2OA_BEFORES	
2	HEX	1489	SZ_TRP_2OA_EXIT	
2	HEX	1490	SZ_TRP_2OD_ENTRY	
2	HEX	1491	SZ_TRP_2OD_BEFORER	
2	HEX	1492	SZ_TRP_2OD_BEFOREP	
2	HEX	1494	SZ_TRP_2OD_EXIT	
2	HEX	1495	SZ_TRP_2OR_ENTRY	
2	HEX	1496	SZ_TRP_2OR_BEFOREP	
2	HEX	1497	SZ_TRP_2OR_EXIT	
2	HEX	1498	SZ_TRP_PSB_ENTRY	
2	HEX	1499	SZ_TRP_PSB_BEFOREO	
2	HEX	149A	SZ_TRP_PSB_BEFORES	
2	HEX	149B	SZ_TRP_PSB_EXIT	
2	HEX	149C	SZ_TRP_PSC_ENTRY	
2	HEX	149D	SZ_TRP_PSC_EXIT	
2	HEX	1502	SZ_TRP_PSD_ENTRY	
2	HEX	1503	SZ_TRP_PSD_BEFORES	
2	HEX	1504	SZ_TRP_PSD_BEFOREP	
2	HEX	1505	SZ_TRP_PSD_EXIT	
2	HEX	1506	SZ_TRP_PSS_ENTRY	
2	HEX	1507	SZ_TRP_PSS_BEFORES	
2	HEX	1508	SZ_TRP_PSS_BEFOREP	
2	HEX	1509	SZ_TRP_PSS_EXIT	
2	HEX	1510	SZ_TRP_PID_ENTRY	
2	HEX	1511	SZ_TRP_PID_BEFORES	
2	HEX	1512	SZ_TRP_PID_BEFOREP	
2	HEX	1513	SZ_TRP_PID_EXIT	
2	HEX	1514	SZ_TRP_POA_ENTRY	
2	HEX	1515	SZ_TRP_POA_BEFORES	
2	HEX	1516	SZ_TRP_POA_EXIT	
2	HEX	1517	SZ_TRP_POD_ENTRY	
2	HEX	1518	SZ_TRP_POD_BEFORER	
2	HEX	1519	SZ_TRP_POD_BEFOREP	
2	HEX	1520	SZ_TRP_POD_BEFORES	
2	HEX	1521	SZ_TRP_POD_EXIT	
2	HEX	1522	SZ_TRP_POR_ENTRY	
2	HEX	1523	SZ_TRP_POR_BEFOREP	
2	HEX	1524	SZ_TRP_POR_EXIT	
2	HEX	1528	SZ_TRP_2SH_ENTRY	
2	HEX	1529	SZ_TRP_2SH_BEFORES	
2	HEX	1530	SZ_TRP_2SH_EXIT	
2	HEX	1531	SZ_TRP_2SQ_ENTRY	
2	HEX	1532	SZ_TRP_2SQ_BEFORES	
2	HEX	1533	SZ_TRP_2SQ_EXIT	
2	HEX	1534	SZ_TRP_2SR_ENTRY	
2	HEX	1535	SZ_TRP_2SR_EXIT	
2	HEX	1536	SZ_TRP_2TE_ENTRY	
2	HEX	1537	SZ_TRP_2TE_BEFORES	
2	HEX	1538	SZ_TRP_2TE_EXIT	
2	HEX	1542	SZ_TRP_PSH_ENTRY	
2	HEX	1543	SZ_TRP_PSH_BEFORES	
2	HEX	1544	SZ_TRP_PSH_EXIT	
2	HEX	1545	SZ_TRP_PSQ_ENTRY	
2	HEX	1546	SZ_TRP_PSQ_BEFORES	
2	HEX	1547	SZ_TRP_PSQ_EXIT	
2	HEX	1548	SZ_TRP_PSR_ENTRY	
2	HEX	1549	SZ_TRP_PSR_EXIT	
2	HEX	1550	SZ_TRP_PTE_ENTRY	

Offset Hex	Type	Len	Name (Dim)	Description
2	HEX	1551	SZ_TRP_PTE_BEFORES	
2	HEX	1552	SZ_TRP_PTE_EXIT	
2	HEX	1553	SZ_TRP_2QS_ENTRY	
2	HEX	1554	SZ_TRP_2QS_EXIT	
2	HEX	1555	SZ_TRP_PQS_ENTRY	
2	HEX	1556	SZ_TRP_PQS_EXIT	
2	HEX	1557	SZ_TRP_BCL_ENTRY	
2	HEX	1558	SZ_TRP_BCL_BEFOREP	
2	HEX	1559	SZ_TRP_BCL_EXIT	
2	HEX	1560	SZ_TRP_BST_ENTRY	
2	HEX	1561	SZ_TRP_BST_GETMAIN	
2	HEX	1562	SZ_TRP_BST_EXIT	
2	HEX	1563	SZ_TRP_BSI_ENTRY	
2	HEX	1564	SZ_TRP_BSI_GETMAIN	
2	HEX	1565	SZ_TRP_BSI_EXIT	
2	HEX	1566	SZ_TRP_BUN_ENTRY	
2	HEX	1567	SZ_TRP_BUN_GETMAIN	
2	HEX	1568	SZ_TRP_BUN_EXIT	
2	HEX	1569	SZ_TRP_BLO_ENTRY	
2	HEX	1570	SZ_TRP_BLO_GETMAIN	
2	HEX	1571	SZ_TRP_BLO_EXIT	
2	HEX	1572	SZ_TRP_VBN_ENTRY	
2	HEX	1573	SZ_TRP_VBN_EXIT	
2	HEX	1576	SZ_TRP_RIA_GETMAIN	
2	HEX	1577	SZ_TRP_RIA_ FREEMAIN	
2	HEX	1578	SZ_TRP_RIQ_GETMAIN	
2	HEX	1579	SZ_TRP_RIQ_FREE	
2	HEX	157A	SZ_TRP_RIF_GETMAIN	
2	HEX	157B	SZ_TRP_RIF_ FREEMAIN	
2	HEX	157C	SZ_TRP_VRI_ENTRY	
2	HEX	157D	SZ_TRP_VRI_BEFORER	
2	HEX	157E	SZ_TRP_VRI_EXIT	
2	HEX	1580	SZ_TRP_VSL_ENTRY	
2	HEX	1581	SZ_TRP_VSL_BEFORES	
2	HEX	1582	SZ_TRP_VSL_EXIT	
2	HEX	1583	SZ_TRP_RPM_ENTRY	
2	HEX	1584	SZ_TRP_RPM_EXIT	
2	HEX	1585	SZ_TRP_RST_ENTRY	
2	HEX	1586	SZ_TRP_RST_EXIT	
2	HEX	1587	SZ_TRP_RTM_ENTRY	
2	HEX	1588	SZ_TRP_RTM_EXIT	
2	HEX	1589	SZ_TRP_RFC_ENTRY	
2	HEX	158A	SZ_TRP_RFC_EXIT	
2	HEX	158B	SZ_TRP_RFC_GETMAIN	
2	HEX	158C	SZ_TRP_RFC_FREE	
2	HEX	158D	SZ_TRP_BSI_ FREEMAIN	
2	HEX	158E	SZ_TRP_BUN_ FREEMAIN	
2	HEX	158F	SZ_TRP_BST_ FREEMAIN	
2	HEX	1590	SZ_TRP_RPM_FREE	
2	HEX	1591	SZ_TRP_2OD_GETMAIN	
2	HEX	1592	SZ_TRP_RIC_FREE	
2	HEX	1593	SZ_TRP_2SB_GETMAIN	
2	HEX	1594	SZ_TRP_2SB_FREE	
2	HEX	1595	SZ_TRP_FSD_ENTRY	
2	HEX	1596	SZ_TRP_FSD_GETMAIN	
2	HEX	1597	SZ_TRP_FSD_EXIT	
2	HEX	1598	SZ_TRP_FRD_ENTRY	
2	HEX	1599	SZ_TRP_FRD_EXIT	
2	HEX	159A	SZ_TRP_BFT_ENTRY	
2	HEX	159B	SZ_TRP_BFT_GETMAIN	
2	HEX	159C	SZ_TRP_BFT_ FREEMAIN	
2	HEX	159D	SZ_TRP_BFT_EXIT	
2	HEX	159E	SZ_TRP_RPM_BADTRAN	
2	HEX	159F	SZ_TRP_BFT_STGERR	
2	HEX	15A0	SZ_TRP_BSI_STGERR1	
2	HEX	15A1	SZ_TRP_BSI_STGERR2	
2	HEX	15A2	SZ_TRP_BST_STGERR1	
2	HEX	15A3	SZ_TRP_BST_STGERR2	
2	HEX	15A4	SZ_TRP_BUN_STGERR1	
2	HEX	15A5	SZ_TRP_BUN_STGERR2	
2	HEX	15A6	SZ_TRP_PSC_FREE	
2	HEX	15A7	SZ_TRP_2SC_FREE	
2	HEX	15A8	SZ_TRP_RST_GETMAIN	
2	HEX	15A9	SZ_TRP_RIC_GETFAIL	
2	HEX	15AA	SZ_TRP_RIO_GETDAC	
2	HEX	15AB	SZ_TRP_RIO_GETTDQ	
2	HEX	15AC	SZ_TRP_RDS_GETMAIN	
2	HEX	15AD	SZ_TRP_RDN_GETMAIN	
2	HEX	15AE	SZ_TRP_RDG_GETMAIN	
2	HEX	15AF	SZ_TRP_RDT_GETMAIN	
2	HEX	15B0	SZ_TRP_POD_GETMAIN	
2	HEX	15B1	SZ_TRP_RCA_GETMAIN	
2	HEX	15B2	SZ_TRP_FSD_FREE	
2	HEX	15B3	SZ_TRP_RIW_GETMAIN	
2	HEX	15B4	SZ_TRP_POR_GETMAIN	
2	HEX	15B5	SZ_TRP_2OR_GETMAIN	

Offset Hex	Type	Len	Name (Dim)	Description
2	HEX	15B6	SZ_TRP_BCS_ENTRY	
2	HEX	15B7	SZ_TRP_BCS_EXIT	
2	HEX	15B8	SZ_TRP_BRS_ENTRY	
2	HEX	15B9	SZ_TRP_BRS_EXIT	
2	HEX	15BA	SZ_TRP_BUS_ENTRY	
2	HEX	15BB	SZ_TRP_BUS_EXIT	
2	HEX	15BC	SZ_TRP_BUS_ GET_FAIL	
2	HEX	15C0	SZ_TRP_IDX_ENTRY	
2	HEX	15C1	SZ_TRP_IDX_EXIT	
2	HEX	15C2	SZ_TRP_IDX_ GET_FAIL	
2	HEX	15C3	SZ_TRP_REQ_ENTRY	
2	HEX	15C4	SZ_TRP_REQ_EXIT	
2	HEX	15C5	SZ_TRP_2OD_BEFORED	
2	HEX	15C6	SZ_TRP_2OD_ BEFOREPD	
2	HEX	15C7	SZ_TRP_2OD_ BEFORES1	
2	HEX	15C8	SZ_TRP_2OD_ BEFORES2	
2	HEX	15C9	SZ_TRP_2OD_ BEFORES3	
Message assignments...				
4	DECIMAL	4001	SZ_MSG_SIP_START	
4	DECIMAL	4002	SZ_MSG_SIP_OK	
4	DECIMAL	4003	SZ_MSG_SIP_END	
4	DECIMAL	4004	SZ_MSG_SIP_ERR_SIT	
4	DECIMAL	4005	SZ_MSG_SIP_ ERR_STATE	
4	DECIMAL	4006	SZ_MSG_SIP_ERR_ENQ	
4	DECIMAL	4007	SZ_MSG_SIP_ERR_SP	
4	DECIMAL	4008	SZ_MSG_SIP_ ERR_RUNAWAY	
4	DECIMAL	4009	SZ_MSG_SIP_ERR_CHP	
4	DECIMAL	4010	SZ_MSG_SIP_ ERR_SWOP	
4	DECIMAL	4099	SZ_MSG_SIP_ABENDED	
4	DECIMAL	4011	SZ_MSG_ZNG_ GET_FAIL	
4	DECIMAL	4012	SZ_MSG_ZAG_ GET_FAIL	
4	DECIMAL	4013	SZ_MSG_ZRG_ GET_FAIL	
4	DECIMAL	4014	SZ_MSG_ZFR_ FREE_FAIL	
4	DECIMAL	4015	SZ_MSG_RDP_SHUT	
4	DECIMAL	4101	SZ_MSG_RII_ INS_NODE_OK	
4	DECIMAL	4102	SZ_MSG_RII_ INS_NODE_FAIL	
4	DECIMAL	4103	SZ_MSG_RDN_ DIS_NODE_OK	
4	DECIMAL	4104	SZ_MSG_RID_ DIS_NODE_SCHED	
4	DECIMAL	4105	SZ_MSG_RID_ DIS_NODE_FAIL	
4	DECIMAL	4106	SZ_MSG_RII_ INS_POOL_OK	
4	DECIMAL	4107	SZ_MSG_RII_ INS_POOL_FAIL	
4	DECIMAL	4108	SZ_MSG_RDG_ DIS_POOL_OK	
4	DECIMAL	4109	SZ_MSG_RID_ DIS_POOL_SCHED	
4	DECIMAL	4110	SZ_MSG_RID_ DIS_POOL_FAIL	
4	DECIMAL	4111	SZ_MSG_RII_ INS_TARG_OK	
4	DECIMAL	4112	SZ_MSG_RII_ INS_TARG_FAIL	
4	DECIMAL	4113	SZ_MSG_RDT_ DIS_TARG_OK	
4	DECIMAL	4114	SZ_MSG_RID_ DIS_TARG_SCHED	
4	DECIMAL	4115	SZ_MSG_RID_ DIS_TARG_FAIL	
4	DECIMAL	4116	SZ_MSG_RII_ INS_PROP_OK	
4	DECIMAL	4117	SZ_MSG_RII_ INS_PROP_FAIL	
4	DECIMAL	4118	SZ_MSG_RID_ DIS_PROP_OK	
4	DECIMAL	4119	SZ_MSG_RID_ DIS_PROP_FAIL	
4	DECIMAL	4120	SZ_MSG_RII_ ADD_NODE_OK	
4	DECIMAL	4121	SZ_MSG_RII_ ADD_NODE_FAIL	
4	DECIMAL	4122	SZ_MSG_RID_ DEL_NODE_OK	
4	DECIMAL	4123	SZ_MSG_RID_ DEL_NODE_FAIL	
4	DECIMAL	4124	SZ_MSG_RII_ ADD_TARG_OK	
4	DECIMAL	4125	SZ_MSG_RII_ ADD_TARG_FAIL	
4	DECIMAL	4126	SZ_MSG_RID_ DEL_TARG_OK	
4	DECIMAL	4127	SZ_MSG_RID_ DEL_TARG_FAIL	

Offset Hex	Type	Len	Name (Dim)	Description
4	DECIMAL	4128	SZ_MSG_RID_ DEL_POOL_FAIL	
4	DECIMAL	4151	SZ_MSG_BUN_UNSOL	
4	DECIMAL	4152	SZ_MSG_BSI_BEGSESS	
4	DECIMAL	4153	SZ_MSG_BST_STSN	
4	DECIMAL	4154	SZ_MSG_BLO_ ACQ_ERROR	
4	DECIMAL	4155	SZ_MSG_BLO_ SESS_ERROR	
4	DECIMAL	4156	SZ_MSG_BFT_FREE	
4	DECIMAL	4157	SZ_MSG_BLO_ ACQ_ERRORX	
4	DECIMAL	4158	SZ_MSG_RIO_ ACQ_ERROR	
4	DECIMAL	4159	SZ_MSG_RIO_ ACQ_ERRORX	
4	DECIMAL	4201	SZ_MSG_RIW_ NODE_STATE	
4	DECIMAL	4202	SZ_MSG_RIW_ POOL_STATE	
4	DECIMAL	4203	SZ_MSG_RIW_ TARG_STATE	

FEP02 Adapter resource manager

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	96	DFHSZAI_ARG	
(0)	CHARACTER	16	SZAI_HEAD	
(0)	HALFWORD	2	SZAI_PLISTLEN	
(2)	HALFWORD	2	*	
(4)	FULLWORD	4	SZAI_FORMAT_NO	
(8)	FULLWORD	4	SZAI_VERSION_NO	
(C)	BITSTRING	4	*	
	1...		SZAI_KERNHANDLE	
(C)	BITSTRING	3	*	

64 EXISTENCE BITS ONE PER KEYWORD IN KEYWORD ORDER

Offset Hex	Type	Len	Name (Dim)	Description
(10)	BITSTRING	8	SZAI_EXISTENCE	
	1...		SZAI_FUNCTION_X	
	.1..		*	
	..1.		SZAI_RESPONSE_X	
	...1		SZAI_REASON_X	
	.... 1...		SZAI_REQUEST_ TYPE_X	
	.... .1..		*	
	.... ..1.		SZAI_ELEMENT_ LENGTH_X	
	1		SZAI_QUEUE_ ELEMENT_X	
(11)	1...		SZAI_CHAINTO_X	
	.1..		SZAI_CONVID_X	
	..1.		SZAI_TERMID_X	
	...1		SZAI_TRANID_X	
	.... 1...		SZAI_TASK_ NUMBER_X	
	.... .1..		SZAI_FQCC_X	

ACTUAL KEYWORDS NOW FOLLOW WITH THEIR
RESPECTIVE ENUMERATED TYPES COMMENTED

Offset Hex	Type	Len	Name (Dim)	Description
(18)	UNSIGNED	1	SZAI_FUNCTION	

SZAI_ PREPARE CONSTANT(001)
SZAI_ QUEUE CONSTANT(002)
SZAI_ RELEASE CONSTANT(003)

Offset Hex	Type	Len	Name (Dim)	Description
(19)	CHARACTER	1	*	
(1A)	UNSIGNED	1	SZAI_RESPONSE	

SZAI_ OK CONSTANT(001)
SZAI_ EXCEPTION CONSTANT(002)
SZAI_ DISASTER CONSTANT(003)
SZAI_ INVALID CONSTANT(004)
SZAI_ KERNERROR CONSTANT(005)
SZAI_ PURGED CONSTANT(006)

Offset Hex	Type	Len	Name (Dim)	Description
(1B)	UNSIGNED	1	SZAI_REASON	

Offset Hex	Type	Len	Name (Dim)	Description
	SZAI_ OK CONSTANT(001)			
	SZAI_ PARMLIST_ INVALID CONSTANT(002)			
	SZAI_ CONVID_ INVALID CONSTANT(003)			
	SZAI_ LENGTH_ INVALID CONSTANT(004)			
	SZAI_ ELEMENT_ INVALID CONSTANT(005)			
	SZAI_ REQUEST_ INVALID CONSTANT(006)			
	SZAI_ CHAINTO_ INVALID CONSTANT(007)			
	SZAI_ RM_INACTIVE CONSTANT(008)			
	SZAI_ GETMAIN_ ERROR CONSTANT(009)			
	SZAI_ NO_STORAGE CONSTANT(010)			
	SZAI_ FREEMAIN_ ERROR CONSTANT(011)			
(1C)	UNSIGNED	1	SZAI_REQUEST_ TYPE	
	SZAI_ ALLOCATE CONSTANT(001)			
	SZAI_ DISCARD CONSTANT(002)			
	SZAI_ EXTRACT CONSTANT(003)			
	SZAI_ FREE CONSTANT(004)			
	SZAI_ INQUIRE CONSTANT(005)			
	SZAI_ INSTALL CONSTANT(006)			
	SZAI_ ISSUE CONSTANT(007)			
	SZAI_ NOOP CONSTANT(008)			
	SZAI_ RECEIVE CONSTANT(009)			
	SZAI_ REQUEST CONSTANT(010)			
	SZAI_ SEND CONSTANT(011)			
	SZAI_ SET CONSTANT(012)			
	SZAI_ START CONSTANT(013)			
	SZAI_ TERMINATE CONSTANT(014)			
	SZAI_ COLLECT_ RESTYPE CONSTANT(015)			
	SZAI_ COLLECT_ RESID CONSTANT(016)			
(1D)	CHARACTER	3	*	
(20)	FULLWORD	4	SZAI_ELEMENT_ LENGTH	
(24)	ADDRESS	4	SZAI_QUEUE_ ELEMENT	
(28)	ADDRESS	4	SZAI_CHAINTO	
(2C)	CHARACTER	8	SZAI_CONVID	
(34)	CHARACTER	4	SZAI_TERMID	
(38)	CHARACTER	4	SZAI_TRANID	
(3C)	CHARACTER	4	SZAI_TASK_NUMBER	
(40)	CHARACTER	27	SZAI_FQCC	
(5B)	CHARACTER	5	*	
(60)	CHARACTER		*	

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	SZAI_PREPARE	
1	DECIMAL	2	SZAI_QUEUE	
1	DECIMAL	3	SZAI_RELEASE	
1	DECIMAL	1	SZAI_OK	
1	DECIMAL	2	SZAI_EXCEPTION	
1	DECIMAL	3	SZAI_DISASTER	
1	DECIMAL	4	SZAI_INVALID	
1	DECIMAL	5	SZAI_KERNERROR	
1	DECIMAL	6	SZAI_PURGED	
1	DECIMAL	2	SZAI_PARMLIST_ INVALID	
1	DECIMAL	3	SZAI_CONVID_ INVALID	
1	DECIMAL	4	SZAI_LENGTH_ INVALID	
1	DECIMAL	5	SZAI_ELEMENT_ INVALID	
1	DECIMAL	6	SZAI_REQUEST_ INVALID	
1	DECIMAL	7	SZAI_CHAINTO_ INVALID	
1	DECIMAL	8	SZAI_RM_INACTIVE	
1	DECIMAL	9	SZAI_GETMAIN_ERROR	
1	DECIMAL	10	SZAI_NO_STORAGE	
1	DECIMAL	11	SZAI_FREEMAIN_ ERROR	
1	DECIMAL	1	SZAI_ALLOCATE	
1	DECIMAL	2	SZAI_DISCARD	
1	DECIMAL	3	SZAI_EXTRACT	
1	DECIMAL	4	SZAI_FREE	
1	DECIMAL	5	SZAI_INQUIRE	
1	DECIMAL	6	SZAI_INSTALL	
1	DECIMAL	7	SZAI_ISSUE	
1	DECIMAL	8	SZAI_NOOP	
1	DECIMAL	9	SZAI_RECEIVE	
1	DECIMAL	10	SZAI_REQUEST	
1	DECIMAL	11	SZAI_SEND	
1	DECIMAL	12	SZAI_SET	
1	DECIMAL	13	SZAI_START	
1	DECIMAL	14	SZAI_TERMINATE	
1	DECIMAL	15	SZAI_COLLECT_ RESTYPE	
1	DECIMAL	16	SZAI_COLLECT_RESID	

FEP03 VTAM acb work area

CONTROL BLOCK NAME = DFHSZDAC
DESCRIPTIVE NAME = CICS (FEPI) VTAM ACB Work Area
FUNCTION = Define 24-bit memory requirements for FEPI
 VTAM control blocks.
 1 control block will exist for each active
 VTAM ACB managed by FEPI. The area is released
 whenever the ACB is deactivated.
LIFETIME = Created by DFHSZRIO during INSTALL processing.
 Deleted by DFHSZRCA during node deactivation.
STORAGE CLASS = 24-bit addressable.
LOCATION = Located from the DFHSZDND which describes the
 node to which the VTAM ACB relates. The DFHSZDND
 is chained from the DFHSZDCM.
INNER CONTROL BLOCKS =
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS =
 MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
 DATA AREAS =
 CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
 GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	172	DFHSZDAC	
(0)	CHARACTER	32	SZD_AC_EYE	Eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	ADDRESS	4	SZD_AC_PREV	Previous
(24)	ADDRESS	4	SZD_AC_NEXT	Next
(24)	BITSTRING	4	SZD_AC_CPA	CLOSE parm area
(28)	CHARACTER	12	*	ACB name
(28)	CHARACTER	1	SZD_AC_NAMEL	
(29)	CHARACTER	8	SZD_AC_NAME	
(31)	CHARACTER	3	*	
(34)	CHARACTER	12	*	ACB password
(34)	CHARACTER	1	SZD_AC_PASSL	
(35)	CHARACTER	8	SZD_AC_PASSWORD	
(3D)	CHARACTER	3	*	
(40)	CHARACTER	108	SZD_AC_ACB	Imbedded VTAM ACB

Constants

Len	Type	Value	Name	Description
4	DECIMAL	172	DFHSZDAC_LEN	

FEP04 Bind request save area

```
CONTROL BLOCK NAME = DFHSZDBI
DESCRIPTIVE NAME = CICS (FEPI) BIND Request Save Area
FUNCTION =
     Defines the BIND Request Save Area.
     This data area is a part of the FEPI Resource Manager.
     It defines the format of the Bind Request Save Area which
     is used when a BIND is received by the SCIP exit and a
     Connection Block is not yet available.
Lifetime = Until OPNSEC can be completed
Storage class = 31-bit addressable
Location = Chained from a Node block
Inner control blocks = Not applicable
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS = None
 MODULE TYPE = Data Area
EXTERNAL REFERENCES:
  DATA AREAS = None
  CONTROL BLOCKS = None
  GLOBAL VARIABLES = None
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	70	DFHSZDBI	
(0)	CHARACTER	32	SZD_BI_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	CHARACTER	8	SZD_BI_WE	SC WE
(20)	BITSTRING	8	SZD_BI_QCB	QCB
(20)	ADDRESS	4	SZD_BI_QC	NEXT ENTRY
(24)	ADDRESS	4	*	Unused
(28)	BITSTRING	4	SZD_BI_FLAGS	
	1...		SZD_BI_DELETED	Logically deleted
	.1..		SZD_BI_REPORT	Reported
(2C)	FULLWORD	4	SZD_BI_CID	CID for the session
(30)	ADDRESS	4	SZD_BI_BINDAREA	ADDRESS OF BIND RU
(34)	FULLWORD	4	SZD_BI_BINDLTH	LENGTH OF BIND RU
(38)	ADDRESS	4	SZD_BI_PARMSESS	ADDRESS OF SESSION PARMS
(3C)	HALFWORD	2	SZD_BI_I_SEQNO	CURRENT REQUESTS SEQ NBR
(3E)	CHARACTER	8	SZD_BI_PRIMARY_LU_NAME	Name of Primary LU

Constants

Len	Type	Value	Name	Description
4	DECIMAL	70	DFHSZDBI_LEN	

FEP05 Connection descriptor

CONTROL BLOCK NAME = DFHSZDCD
DESCRIPTIVE NAME = CICS (FEPI) Connection Descriptor
FUNCTION = Represents a connection to the resource manager.
Contains all of the information and references
needed by the resource manager to manage a network
connection between the front-end node and the
back-end target system.
LIFETIME = Created by DFHSZRIC during INSTALL processing.
Deleted by DFHSZRDC during DISCARD processing.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDPD which describes the
pool to which the connection belongs. The DFHSZDPD
is chained from the DFHSZDCM.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	380	DFHSZDCD	
(0)	CHARACTER	32	SZD_CD_EYE	Eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	CHARACTER	24	SZD_CD_SC_WE	SC DQE
(20)	BITSTRING	8	SZD_CD_SC_QCB	SC DQE
(20)	ADDRESS	4	SZD_CD_SC_QP	Prev Q'd element
(24)	ADDRESS	4	SZD_CD_SC_QC	Next Q'd element
(28)	FULLWORD	4	SZD_CD_SC_REQ	Request type
(2C)	BITSTRING	4	*	Request flags
	1...		*	Reserved - not avail
	.1..		*	Reserved - not avail
	..1.		SZD_CD_ON_SCQ	On the process Q
	...1		SZD_CD_ ON_SCQIRB	On the IRB process Q
	.... 1...		SZD_CD_ON_TMR	Reserved - not avail
	.... .1..		*	Reserved - not avail

NOTE
End of portion that must match DFHSZDQE. The following 2 fields are identically placed in node, targets and conn's.

Offset Hex	Type	Len	Name (Dim)	Description
(30)	HALFWORD	2	SZD_CD_TRINTVL	Timer retry interval
(32)	HALFWORD	2	SZD_CD_TRTYPE	Retry type required
(34)	FULLWORD	4	*	Unused available

These portion is used for queuing the connection to a target for REQSESS processing.

Offset Hex	Type	Len	Name (Dim)	Description
(38)	CHARACTER	12	SZD_CD_RE_WE	RE WE
(38)	BITSTRING	8	SZD_CD_RE_QCB	RE QCB
(38)	ADDRESS	4	SZD_CD_RE_QC	Next entry
(3C)	ADDRESS	4	*	Unused
(40)	FULLWORD	4	SZD_CD_RE_REQ	Request type

Connection control flags
This word (SZD_CD_FLAGS_ALLOC) is tested for zero. A value of zero indicates that the connection is OK to be allocated. Therefore, all flags in this word must be such that one makes the connection unavailable for use.

Offset Hex	Type	Len	Name (Dim)	Description
(44)	BITSTRING	4	SZD_CD_ FLAGS_ALLOC	
(44)	BITSTRING	1	SZD_CD_ FLAGS_ALLOC1	
				Data Traffic Reset
	1...		SZD_CD_DTR	Data Traffic Reset
	.111 111.		*	Unused available @BA70191C
	1		SZD_CD_TERM_Q	TERMQ flag @BA70191A
(45)	BITSTRING	1	SZD_CD_ FLAGS_ALLOC2	
	1...		SZD_CD_LOST	Session lost
	.1..		SZD_CD_LOFF	Session failed drop it
	..1.		SZD_CD_SHUTD	SHUTD Received
	...1		SZD_CD_TERM_U	Termination requested Unconditionally
	.... 1...		SZD_CD_TERM_C	Termination requested Conditionally
	.... .1..		SZD_CD_QEC	QEC Received

Offset Hex	Type	Len	Name (Dim)	Description
	.... ..1.		SZD_CD_ DRAINING	Draining session
	1		SZD_CD_ PEND_MORNING	Good Morning pending
(46)	BITSTRING	1	SZD_CD_ FLAGS_ALLOC3	
	1...		SZD_CD_ALLOC	Connection in use
	.1..		SZD_CD_ POS_DRAINING	+ve draining @BA59262C
	..11 1111		*	Unused @BA59262A
(47)	BITSTRING	1	SZD_CD_ FLAGS_ALLOC4	
(47)	BITSTRING	1	*	unused - available
(48)	BITSTRING	1	SZD_CD_FLAGS_SC1	
	1...		SZD_CD_QC	QC Sent
	.1..		SZD_CD_RELQ	RELQ Received
	..1.		SZD_CD_INB	IN BRACKET
	...1		SZD_CD_CD_SENT	CD Sent
	.... 1...		SZD_CD_MIC	First in chain sent
	.... .1..		SZD_CD_SDTR	SDT Received
	.... ..1.		SZD_CD_PEND_EB	Pending EB
	1		SZD_CD_ AWAITING_RESPONSE	API Receive posted
(49)	BITSTRING	1	SZD_CD_FLAGS_SC2	
	1...		SZD_CD_ RCVD_MORNING	Good Morning Received
	.1..		SZD_CD_ BID_PURGE	BID PURGE
	..11 1111		*	UNUSED - AVAIL
(4A)	BITSTRING	1	SZD_CD_FLAGS_SS1	Session state
	1...		SZD_CD_CLEARR	CLEAR Received Presentation space lost if LU2
	.1..		SZD_CD_CLEARREP	CLEAR reported
	..1.		SZD_CD_SIP	SEND in progress
	...1		*	unused available
	.... 1...		SZD_CD_SHUTC	SHUTC Sent
	.... .1..		SZD_CD_UNBINDR	UNBIND Received
	.... ..1.		SZD_CD_NSEXITR	NSEXIT Scheduled
	1		SZD_CD_LOSTR	Failure reported
(4B)	BITSTRING	1	SZD_CD_FLAGS_SS2	Session state
	1...		SZD_CD_OPNSEC	OPNSEC ISSUED
	.1..		SZD_CD_ OPNSEC_OK	OPNSEC Accepted
	..1.		SZD_CD_ OPNSEC_REJ	OPNSEC REJECTED
	...1		*	unused available
	.... 1...		SZD_CD_STSN	STSN PROCESSED
	.... .1..		SZD_CD_STSN_OK	STSN Response Accepted
	.... ..1.		SZD_CD_ STSN_SCHED	STSN Transaction Start
	1		SZD_CD_STSNR	STSN Received
(4C)	BITSTRING	1	SZD_CD_FLAGS_SS3	Session state
	1...		SZD_CD_SDT_OK	SDT Response Accepted
	.1..		SZD_CD_SDT_REP	SDT Response Initiated
	..1.		SZD_CD_ BSX_SCHED	Beginsession exit sched
	...1		SZD_CD_ UDX_SCHED	Unsol. data exit sched
	.... 1...		SZD_CD_REQ	REQSESS ISSUED
	.... .1..		SZD_CD_REQD	REQSESS Accepted
	.... ..1.		SZD_CD_ FSX_SCHED	FREE exit scheduled
	1		*	Unused
(4D)	1111		SZD_CD_FLAGS_PP1	Property flags
	1...		SZD_CD_XCPTN_X	Exception xactn exists
	.1..		SZD_CD_STSN_X	STSN xactn exists
	..1.		SZD_CD_SIGNON_X	SIGNON xactn exists
	...1		SZD_CD_UNSOLD_X	Unsolictd xactn exists
	 1111		SZD_CD_FLAGS_FP1	FREE processing flags
	.... 1...		SZD_CD_FREEQD	API FREE requested
	.... .1..		SZD_CD_FREEF	FREE force
	.... ..1.		SZD_CD_FREER	FREE release
	1		SZD_CD_AGATE	API queuing gate
(4E)	BITSTRING	1	SZD_CD_ FLAGS_TTD1	
	1...		SZD_CD_ USX_SCHED	Unbind xaction sched'd
	.1..		SZD_CD_ SDX_SCHED	Start data xaction schd
	..1.		SZD_CD_ON_REQ	ON THE REQSESS Q
	...1		SZD_CD_ ON_REQIRB	ON THE REQSESS Q
	.... 1...		SZD_CD_BINDR	BIND Received
	.... .1..		SZD_CD_PENDTR	Xaction pending
	.... ..1.		SZD_CD_DATAR	REC(ANY) Data Received
	1		SZD_CD_RESPR	REC(ANY) RESP Received
(4F)	BITSTRING	1	SZD_CD_ FLAGS_TTD2	Misc flags @BA83689C
	1...		SZD_CD_NDCLOSE	Node is closing
	.1..		SZD_CD_ API_QUEUED	API request queued
	..1.		SZD_CD_ GOOD_MORNING	Good Morning expected
	...1		SZD_CD_LOSE	Lose contention
	.... 1...		SZD_CD_FREE_X	Free exit supplied
	.... .1..		SZD_CD_UDFLAG	Unsol tracking
	.... ..1.		SZD_CD_URFLAG	Unsol tracking
	1		SZD_CD_DYNAM	Dynamic session

Offset Hex	Type	Len	Name (Dim)	Description

These flags allow DFHSZRDC to determine what additional cleanup may be required when this connection is removed. Each flag identifies a parent node whose deletion is pending the removal of all of the connections to which it relates. CONN is always set if a connection is being deleted. One or all of the other bits may be set.

(50)	BITSTRING	1	SZD_CD_DREASON	Discard reason codes
	1...		SZD_CD_DEL_CONN	Connection deleted
	.1..		SZD_CD_DEL_NODE	NODE discarded
	..1.		SZD_CD_DEL_POOL	Pool discarded
	...1		SZD_CD_ DEL_TARGET	Target discarded
	 1111		*	Unused available
(51)	BITSTRING	1	SZD_CD_MISC	Miscellaneous flags
	1...		SZD_CD_EXREQ	External BIND requested
	.1..		SZD_CD_ ALLOC_INC	CD is allocated
	..11 1111		*	Unused available
(52)	BITSTRING	2	*	Unused available

Connection information

(54)	ADDRESS	4	SZD_CD_DATA_DRA	Data Receive DRA
(58)	ADDRESS	4	SZD_CD_RESP_DRA	Resp Receive DRA
(5C)	ADDRESS	4	SZD_CD_BINDAREA	Address of BIND RU
(60)	ADDRESS	4	SZD_CD_API_QE	API QE pointer
(64)	ADDRESS	4	SZD_CD_PARMSESS	Address of session parms
(68)	FULLWORD	4	SZD_CD_CID	CID for the session
(6C)	FULLWORD	4	SZD_CD_BINDLTH	LENGTH OF BIND RU
(70)	FULLWORD	4	SZD_CD_ EVENTVALUE	EVENTVALUE for lost Session
(74)	HALFWORD	2	SZD_CD_DEVICE	Device type token
(76)	UNSIGNED	2	SZD_CD_IBSQVAL	Inbound sequence nbr
(78)	UNSIGNED	2	SZD_CD_OBSQVAL	Outbound sequence nbr
(7A)	BITSTRING	1	SZD_CD_IBSQAC	Inbound SET/TESTSET
(7B)	BITSTRING	1	SZD_CD_OBSQAC	Outbound SET/TESTSET
(7C)	UNSIGNED	2	SZD_CD_I_SEQNO	Current requests seq nbr
(7E)	UNSIGNED	2	SZD_CD_O_SEQNO	Latest Hostbound seq nbr
(80)	UNSIGNED	2	SZD_CD_RETCODE	Return code from Receive CHECK processing
(82)	HALFWORD	2	SZD_CD_ UNBIND_LTH	UNBIND code length
(84)	HALFWORD	2	SZD_CD_ NSEXIT_LTH	NSEXIT code length
(86)	HALFWORD	2	*	padding
(88)	CHARACTER	4	SZD_CD_ UNBIND_CODE	UNBIND code
(8C)	CHARACTER	32	SZD_CD_ NSEXIT_CODE	NSEXIT code
(AC)	CHARACTER	8	SZD_CD_LOGMODE	LOGMODE name
(B4)	CHARACTER	4	SZD_CD_TDQ	TDQ name
(B8)	CHARACTER	4	SZD_CD_ SIGNON_TRAN	SIGNON xactn name
(BC)	CHARACTER	4	SZD_CD_STSN_TRAN	STSN xactn name
(C0)	CHARACTER	4	SZD_CD_ UNSOL_TRAN	Unsolicited data xactn

Configuration control information.
A connection exists on three lists:
(1) NEXT/PREV chain the connection to the pool which owns it
(2) NDNEXT/NDPREV chains it to the node on which it depends
(3) TDNEXT/TDPREV chains it to the target on which it depends

(C4)	CHARACTER	160	SZD_CD_API	
(C4)	ADDRESS	4	SZD_CD_PREV	DPD chain area
(C8)	ADDRESS	4	SZD_CD_NEXT	
(CC)	ADDRESS	4	SZD_CD_NDPREV	DND chain area
(D0)	ADDRESS	4	SZD_CD_NDNEXT	
(D4)	ADDRESS	4	SZD_CD_TDPREV	DTD chain area
(D8)	ADDRESS	4	SZD_CD_TDNEXT	
(DC)	ADDRESS	4	SZD_CD_PDPTR	associated DPD
(E0)	ADDRESS	4	SZD_CD_TDPTR	associated DTD
(E4)	ADDRESS	4	SZD_CD_NDPTR	associated DND
(E8)	ADDRESS	4	SZD_CD_CVPTR	associated DCV
(EC)	HALFWORD	2	SZD_CD_ SERVSTATUS	Service status
(EE)	HALFWORD	2	SZD_CD_ ACQSTATUS	Network status actual
(F0)	HALFWORD	2	SZD_CD_ DESSTATUS	Network status desired
(F2)	HALFWORD	2	SZD_CD_ INSTSTATUS	Installation status
(F4)	HALFWORD	2	SZD_CD_ SESSSTATUS	Session status
(F6)	HALFWORD	2	*	Unused available
(F8)	FULLWORD	4	SZD_CD_CURRENT	Usage counter
(FC)	FULLWORD	4	SZD_CD_USAGE	Usage counter
(100)	ADDRESS	4	SZD_CD_DSPTR	Fmt extension
(104)	ADDRESS	4	SZD_CD_DCPREV	Dump chain
(108)	ADDRESS	4	SZD_CD_DCNEXT	Dump chain
(10C)	CHARACTER	4	SZD_CD_ FREE_TRAN	FREE exit
(110)	FULLWORD	4	SZD_CD_USENSE	User sense
(114)	FULLWORD	4	SZD_CD_SSENSE	System Sense
(118)	ADDRESS	4	SZD_CD_RDPTR	Buffer address
(11C)	FULLWORD	4	SZD_CD_RDLEN	Buffer length
(120)	FULLWORD	4	SZD_CD_RCOUNT	Retry count
(124)	CHARACTER	64	SZD_CD_UDATA	User data

Statistics counters

(164)	FULLWORD	4	SZD_CD_SENT	# characters sent on connection
(168)	FULLWORD	4	SZD_CD_RECEIVED	# characters received on connection

Offset Hex	Type	Len	Name (Dim)	Description
(16C)	FULLWORD	4	SZD_CD_ UNSOLICITEDINPUTS	# unsolicited inputs on connection
(170)	FULLWORD	4	SZD_CD_ RECEIVETIMEOUTS	# RECEIVEs that timed out
(174)	FULLWORD	4	SZD_CD_ERRORS	# Error conditions
(178)	FULLWORD	4	SZD_CD_END	Structure end *

Constants

Len	Type	Value	Name	Description
4	DECIMAL	380	DFHSZDCD_LEN	

FEP06 Common data area

CONTROL BLOCK NAME = DFHSZDCM
DESCRIPTIVE NAME = CICS (FEPI) Common data area
FUNCTION = Base FEPI resource manager data area from which
all other FEPI data areas may be located. Also
contains all globally referenced single instance
data areas. There is one DFHSZDCM.
LIFETIME = Obtained by DFHSZSIP during resource manager
initialisation. Released by DFHSZSIP during
resource manager termination.
STORAGE CLASS = 31-bit RW
LOCATION = Addressed by DFHSZSDS static area structure.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	432	DFHSZDCM	
(0)	CHARACTER	32	SZD_CM_EYE	Eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
Dispatcher work Q anchors (1)				
(20)	BITSTRING	4	SZD_CM_SC_QCB	PRB normal reqs
(20)	ADDRESS	4	SZD_CM_SC_QC	External anchor
(24)	ADDRESS	4	SZD_CM_SC_SYS	Internal anchor
(28)	BITSTRING	4	SZD_CM_SC_QCBT	PRB timed reqs
(28)	ADDRESS	4	SZD_CM_SC_QCT	External anchor
(2C)	ADDRESS	4	SZD_CM_SC_SYST	Internal anchor
(30)	BITSTRING	4	SZD_CM_ SC_QCBIRBT	IRB timed reqs
(30)	ADDRESS	4	SZD_CM_ SC_QCIRBT	External anchor
(34)	ADDRESS	4	SZD_CM_ SC_SYSIRBT	Internal anchor
(38)	BITSTRING	4	SZD_CM_SC_QCBIRB	IRB normal reqs
(38)	ADDRESS	4	SZD_CM_SC_QCIRB	External anchor
(3C)	ADDRESS	4	SZD_CM_SC_SYSIRB	Internal anchor
(40)	BITSTRING	4	SZD_CM_ SC_QCBTPEND8	IRB TPEND8 reqs
(40)	ADDRESS	4	SZD_CM_ SC_QCTPEND8	External anchor
(44)	ADDRESS	4	SZD_CM_ SC_SYSTPEND8	Internal anchor
VTAM IRB request work areas				
(48)	BITSTRING	4	SZD_CM_FREE_QCB	Free RB queue
(48)	ADDRESS	4	SZD_CM_ FREE_QUEUE	FIRST ENTRY
(4C)	ADDRESS	4	SZD_CM_IRBSAVE	IRB LIFO stack area
(50)	ADDRESS	4	SZD_CM_RPL_MASK	standard RPL mask address

Offset Hex	Type	Len	Name (Dim)	Description
(54)	ADDRESS	4	SZD_CM_ OPNSEC_MASK	OPNSEC mask address
(58)	ADDRESS	4	SZD_CM_ RECANY_MASK	RECEIVE(ANY) mask address
(5C)	ADDRESS	4	SZD_CM_NIB_MASK	NIB mask address
Resource manager miscellaneous				
(60)	ADDRESS	4	SZD_CM_LIFO	RM LIFO stack base
(64)	ADDRESS	4	SZD_CM_ ACTIVE_CVLIST	Active conversations
(68)	ADDRESS	4	SZD_CM_ INACTIVE_CVLIST	Inactive conversations
(6C)	ADDRESS	4	SZD_CM_NDLIST	System node list
(70)	ADDRESS	4	SZD_CM_TDLIST	System target list
(74)	ADDRESS	4	SZD_CM_PDLIST	System pool list
(78)	ADDRESS	4	SZD_CM_PSLIST	Property set list
(7C)	ADDRESS	4	SZD_CM_CQE	Current DQE
(80)	ADDRESS	4	SZD_CM_TQE	Terminate DQE
(84)	ADDRESS	4	SZD_CM_SDS	Static area address
(88)	ADDRESS	4	SZD_CM_EXLST	VTAM EXLST address
(8C)	ADDRESS	4	SZD_CM_ACBTEMP	OPEN work queue
(90)	HALFWORD	2	SZD_CM_DSTAT	Dispatcher status
(92)	BITSTRING	2	SZD_CM_FLAGS	
	1...		SZD_CM_SCHEDPPM	TDQ/IC trigger
	.1..		SZD_CM_SCHEDTQA	Recovery trigger
	..1.		SZD_CM_STIMFAIL	STIMERM fail@BA72241A
(94)	FULLWORD	4	SZD_CM_WAITK	Disp. WAIT counter
(98)	FULLWORD	4	SZD_CM_RASIZE	REC(ANY) buffer size
(9C)	ADDRESS	4	SZD_CM_BCLIST	BROWSE list anchor
(A0)	ADDRESS	4	SZD_CM_TOLIST	Timed request anchor
(A4)	FULLWORD	4	SZD_CM_TICK	Timer tick
(A8)	FULLWORD	4	SZD_CM_DISPK	Dispatch counter
(AC)	FULLWORD	4	SZD_CM_DDLIST	Deferred discard q
CICS environment save area				
(B0)	ADDRESS	4	SZD_KESTACK_SAVE	CICS stack pointer
(B4)	ADDRESS	4	SZD_TCA_SAVE	CICS TCA address
(B8)	CHARACTER	64	SZD_REGS_SAVE	CICS registers
Dispatcher ECB list for DSSRWAIT				
(F8)	CHARACTER	88	SZD_CM_QECBLIST	
(F8)	ADDRESS	4	SZD_CM_EQPTR	Expedited Q ECB address
(FC)	ADDRESS	4	SZD_CM_XQPTR	Unused Q ECB address
(100)	ADDRESS	4	SZD_CM_CQPTR	Unused Q ECB address
(104)	ADDRESS	4	SZD_CM_IQPTR	API inbound Q ECB address
(108)	ADDRESS	4	SZD_CM_ SC_PTRIRB	IRB normal ECB address
(10C)	ADDRESS	4	SZD_CM_ SC_PTRIRBT	IRB timer ECB address
(110)	ADDRESS	4	SZD_CM_ SC_PTRTPEND8	IRB TPEND8 ECB address
Dispatcher work queue ECBs				
(114)	BITSTRING	4	SZD_CM_EQECB	
(118)	BITSTRING	4	SZD_CM_XQECB	
(11C)	BITSTRING	4	SZD_CM_CQECB	
(120)	BITSTRING	4	SZD_CM_IQECB	
(124)	ADDRESS	4	SZD_CM_ SC_ECBIRB	
(128)	ADDRESS	4	SZD_CM_ SC_ECBIRBT	
(12C)	ADDRESS	4	SZD_CM_ SC_ECBTPEND8	
Dispatcher work q anchors (2)				
(130)	ADDRESS	4	SZD_CM_EQHEAD	Expedited requests
(134)	ADDRESS	4	SZD_CM_EQSYS	
(138)	ADDRESS	4	SZD_CM_XQHEAD	TDQ/START request Q
(13C)	ADDRESS	4	SZD_CM_XQSYS	
(140)	ADDRESS	4	SZD_CM_CQHEAD	Unused
(144)	ADDRESS	4	SZD_CM_CQSYS	
(148)	ADDRESS	4	SZD_CM_IQHEAD	API PRB queue header
(14C)	ADDRESS	4	SZD_CM_IQSYS	
STIMERM work area				
(150)	CHARACTER	60	SZD_CM_ STIMERM_PARMS	
(150)	FULLWORD	4	SZD_CM_STFLAGS	STIMER flags
(154)	ADDRESS	4	SZD_CM_TICKIDA	Timer ID adress
(158)	ADDRESS	4	SZD_CM_TICKPTR	Timer tick len ptr
(15C)	ADDRESS	4	SZD_CM_STEXIT	Timer exit address
(160)	ADDRESS	4	SZD_CM_STPARM	Timer parm address
(164)	UNSIGNED	4	*	Padding
(168)	FULLWORD	4	SZD_CM_TICKLEN	Timer tick length
(16C)	FULLWORD	4	SZD_CM_TICKID	Timer ID value
TDQ/STQ batch queue anchor				
(170)	FULLWORD	4	SZD_CM_DCQLIST	TD and IC queue
Timed retry work area				
(174)	HALFWORD	2	SZD_CM_RETRY	Retry delay
(176)	HALFWORD	2	SZD_CM_RETRYK	Retry origin

Offset Hex	Type	Len	Name (Dim)	Description
(178)	ADDRESS	4	SZD_CM_TQALIST	Timed recovery Q
	Connection list for dump formatting			
(17C)	ADDRESS	4	SZD_CM_CDLIST	Dump conn. list
	LIFO size constants for dump formatting			
(180)	FULLWORD	4	SZD_CM_IRBLEN	IRB LIFO length
(184)	FULLWORD	4	SZD_CM_LIFOLEN	PRB LIFO length
	VTAM ACB/RPL exit footprints			
(188)	BITSTRING	4	SZD_CM_EXITMSK	IRB exit mask
	1...		SZD_CM_XTP	TPEND
	.1..		SZD_CM_XNS	NSEXIT
	..1.		SZD_CM_XSC	SCIP
	...1		SZD_CM_XLT	LOSTTERM
	.... 1...		SZD_CM_XRA	RECEIVE any
	.... .1..		SZD_CM_XFR	Common RPL
	.... ..1.		SZD_CM_XDA	DFASY
	1		SZD_CM_WSL	SETLOGON RPL
(189)	1...		SZD_CM_2IX	SEND RPL (LU2)
	.1..		SZD_CM_2DX	LU 2 Drain RPL
	..1.		SZD_CM_2OX	LU 2 REC(SPEC)
	...1		SZD_CM_2QX	LU 2 REQSESS RPL
	.... 1...		SZD_CM_2SX	LU 2 OPNSEC
	.... .1..		SZD_CM_2PX	LU 2 +ve drain @BA59262C
	.... ..1.		*	unused - available
	1		*	unused - available
(18A)	1...		SZD_CM_PIX	SEND RPL (LUP)
	.1..		SZD_CM_PDX	LU P Drain RPL
	..1.		SZD_CM_POX	LU P REC(SPEC)
	...1		SZD_CM_PQX	LU P REQSESS
	.... 1...		SZD_CM_PSX	LU P OPNSEC
	.... .1..		*	unused - available
	.... ..1.		*	unused - available
	1		*	unused - available
(18B)	1...		SZD_CM_YQR	REQSESS Queuer
	.1..		SZD_CM_YRI	R(A) issuer
	..1.		SZD_CM_YSC	Unsol. BIND handlr
	...1		SZD_CM_YSR	R(A) feedback int.
	.... 1...		SZD_CM_YSY	IRB feedback int.
	.... .1..		*	unused - available
	.... ..1.		*	unused - available
	1		*	unused - available
	CONVID generation area			
(18C)	FULLWORD	4	SZD_CM_CVID	CONVID memory
(190)	FULLWORD	4	SZD_CM_RMID	CONVID extension
(194)	FULLWORD	4	SZD_CM_RETRY1	Timer retry intvl
(198)	FULLWORD	4	SZD_CM_RETRY2	Timer retry intvl
(19C)	FULLWORD	4	SZD_CM_RLIM	Max retry count
(1A0)	ADDRESS	4	SZD_CM_DDDLIST	delayed DDLIST
(1A4)	CHARACTER	8	SZD_CM_ STIMERM_ECB	STIMERM ECB fields @BA72241A
(1A4)	ADDRESS	4	SZD_CM_STPTR	pointer to ECB @BA72241A
(1A8)	BITSTRING	4	SZD_CM_STECB	ECB @BA72241A
(1AC)	FULLWORD	4	SZD_CM_END	end-of-structure

TDQ request queue element. Processed by RPM every 1s.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	120	SZD_TDQ_QREQ	
(0)	CHARACTER	32	SZD_TDQ_EYE	Eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	ADDRESS	4	SZD_TDQ_QNEXT	next TDQ/STQ on batch q
(24)	CHARACTER	4	TDQ_QUEUER	originating module
(28)	FULLWORD	4	*	
(2C)	CHARACTER	72	TDQDATA	data to be queued
(2C)	FULLWORD	4	TDQ_DATATYPE	
(30)	FULLWORD	4	TDQ_EVENTTYPE	
(34)	FULLWORD	4	TDQ_EVENTVALUE	
(38)	CHARACTER	8	TDQ_EVENTDATA	
(38)	FULLWORD	4	TDQ_EVENT1	
(3C)	FULLWORD	4	TDQ_EVENT2	
(40)	CHARACTER	4	TDQ_SPARE4	
(44)	CHARACTER	8	TDQ_POOL	

Offset Hex	Type	Len	Name (Dim)	Description
(4C)	CHARACTER	8	TDQ_TARGET	
(54)	CHARACTER	8	TDQ_NODE	
(5C)	BITSTRING	8	TDQ_CONVID	
(64)	FULLWORD	4	TDQ_DEVICE	
(68)	FULLWORD	4	TDQ_FORMAT	
(6C)	CHARACTER	8	TDQ_SPARE8	
(74)	CHARACTER	4	TDQ_QUEUE	Target TDQ name

START request queue element. Processed by RPM every 1s.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	256	SZD_STQ_QREQ	
(0)	CHARACTER	32	SZD_STQ_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	ADDRESS	4	SZD_STQ_QNEXT	next STQ onbatching queue
(24)	CHARACTER	4	STQ_QUEUER	originating module
(28)	CHARACTER	208	STQDATA	START data queued by IC
(28)	HALFWORD	2	STQ_DATALENGTH	
(2A)	HALFWORD	2	*	
(2C)	FULLWORD	4	STQ_DATATYPE	
(30)	FULLWORD	4	STQ_EVENTTYPE	
(34)	FULLWORD	4	STQ_EVENTVALUE	
(38)	CHARACTER	8	STQ_EVENTDATA	
(38)	FULLWORD	4	STQ_EVENT1	
(3C)	FULLWORD	4	STQ_EVENT2	
(40)	CHARACTER	4	STQ_SPARE4	
(44)	CHARACTER	8	STQ_POOL	
(4C)	CHARACTER	8	STQ_TARGET	
(54)	CHARACTER	8	STQ_NODE	
(5C)	BITSTRING	8	STQ_CONVID	
(64)	FULLWORD	4	STQ_DEVICE	
(68)	FULLWORD	4	STQ_FORMAT	
(6C)	CHARACTER	8	STQ_SPARE8	
(74)	FULLWORD	4	STQ_FLENGTH	
(78)	CHARACTER	128	STQ_USERDATA	
(F8)	CHARACTER	4	STQ_TRANSID	Transaction to start
(FC)	CHARACTER	4	STQ_TERMID	Terminal to obtain

USS record queue element.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	SZD_USQ_QREQ	
(0)	CHARACTER	32	SZD_USQ_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	ADDRESS	4	SZD_USQ_QNEXT	next USQ onbatching queue
(24)	CHARACTER	4	USQ_QUEUER	originating module
(28)	ADDRESS	4	USQ_RECORD_PTR	->USQ_RECORD
(2C)	CHARACTER	4	USQDATA	USS record:
(2C)	FULLWORD	4	USQ_DATATYPE	Queue element type - 3
(30)	FULLWORD	4	USQ_RECORD	USS record: DFHA22PS - pool DFHA23PS - connection DFHA24PS - target

Install/discard exit queue element.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	75	SZD_IDQ_QREQ	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER	32	SZD_IDQ_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	ADDRESS	4	SZD_IDQ_QNEXT	next IDQ on batch queue
(24)	CHARACTER	8	*	Reserved
(2C)	CHARACTER	31	IDQDATA	XRSINDI parameters
(2C)	FULLWORD	4	IDQ_DATATYPE	Queue element type - 4
(30)	CHARACTER	16	IDQ_RES_NAME	Resource name
(40)	FULLWORD	4	IDQ_NAME_LENGTH	Resource name length
(44)	FULLWORD	4	IDQ_NUMBER	Number of resources
(48)	UNSIGNED	1	IDQ_INSTDISC	Request type identifier
(49)	UNSIGNED	1	IDQ_RES_TYPE	Resource type
(4A)	UNSIGNED	1	IDQ_RECOVERY	Resource recovery

Constants

Len	Type	Value	Name	Description
4	DECIMAL	100	SZK_RSC	Connection
4	DECIMAL	104	SZK_RNC	Node
4	DECIMAL	108	SZK_RTC	Target

Resource manager recovery retry resource types.

4	DECIMAL	110	SZK_RSCT	
4	DECIMAL	114	SZK_RNCT	
4	DECIMAL	118	SZK_RTCT	

Resource manager recovery retry processing types

4	DECIMAL	256	SZK_REOPEN	
4	DECIMAL	257	SZK_REQUEUE	
4	DECIMAL	258	SZK_REISSUE	

Resource manager recognised LU types.

4	DECIMAL	1	SZK_SLU2	
4	DECIMAL	2	SZK_SLUP	

REQSESS EVENTVALUE values Set by 2QX and PQX RPL exits

4	DECIMAL	199	SZK_SFAIL_REQSESS_NOT_AVAIL	
4	DECIMAL	198	SZK_SFAIL_REQSESS_INHIBITED	
4	DECIMAL	197	SZK_SFAIL_REQSESS_OTHER	

NSEXIT EVENTVALUE values Set by XNS ACB exit.

4	DECIMAL	196	SZK_SFAIL_CINIT	NOTIFY
4	DECIMAL	195	SZK_SFAIL_BIND	NOTIFY
4	DECIMAL	194	SZK_SFAIL_PLU	NOTIFY
4	DECIMAL	193	SZK_SFAIL_SLU	NOTIFY
4	DECIMAL	192	SZK_SFAIL_SSCP	NOTIFY
4	DECIMAL	191	SZK_SFAIL_ UNDEF_SETUP	NOTIFY
4	DECIMAL	190	SZK_SLOST_TAKEDOWN	NOTIFY
4	DECIMAL	189	SZK_SLOST_CLEANUP_NORM	CLEANUP
4	DECIMAL	188	SZK_SLOST_CLEANUP_ABNORM	CLEANUP

LOSTERM EVENTVALUE values Set by XLT ACB exit.

4	DECIMAL	187	SZK_SLOST_LOSTERM	LOSTERM

Session control EVENT values Set by XSC ACB exit.

4	DECIMAL	186	SZK_SLOST_UNBIND_NORMAL	
4	DECIMAL	185	SZK_SLOST_ UNBIND_BIND	
4	DECIMAL	184	SZK_SLOST_UNBIND_INVALID	
4	DECIMAL	183	SZK_SLOST_UNBIND_RECOV	
4	DECIMAL	182	SZK_SLOST_UNBIND_UNRECOV	

Resource manager internal constant values

4	DECIMAL	65536	SZK_LIFO_LENGTH	
4	DECIMAL	8192	SZK_IRB_LENGTH	
4	DECIMAL	4096	SZK_RASIZE	
4	DECIMAL	100	SZK_TS_TICKLEN	
0	BIT	1	SZK_FLAG_ON	
0	BIT	0	SZK_FLAG_OFF	

Len	Type	Value	Name	Description
	Resource manager internal return codes			
4	DECIMAL	0	SZK_RC_OK	
4	DECIMAL	4	SZK_RC_NO_STORAGE	
4	DECIMAL	32	SZK_RC_INVREQ	
4	DECIMAL	122	SZK_RC_DEFER	
4	DECIMAL	97	SZK_RC_EMPTY	
4	DECIMAL	98	SZK_RC_POST	
4	DECIMAL	99	SZK_RC_NOPOST	
	Dispatcher (RDP) processing states			
2	DECIMAL	1	SZK_DS_RUN	
2	DECIMAL	2	SZK_DS_WAIT	
2	DECIMAL	3	SZK_DS_INIT	
2	DECIMAL	4	SZK_DS_END	
	ADD processing reason codes			
2	DECIMAL	5	SZK_ADD_NODE	
2	DECIMAL	6	SZK_ADD_TARGET	
	Delete processors result codes @BA73815A @BA73815A @BA73815A			
2	DECIMAL	7	SZK_RDN_NODE_ DELETED	@BA73815A @BA73815A
	Compare-and-Swap condition code equate			
1	DECIMAL	4	SZK_CC_OK	
4	DECIMAL	432	DFHSZDCM_LEN	

FEP07 Conversation data area

CONTROL BLOCK NAME = DFHSZDCV
DESCRIPTIVE NAME = CICS (FEPI) Conversation Data Area
FUNCTION = Contains the information needed by the resource
manager to control an allocated connection
(a conversation). One CVCB will exist for each
allocated connection.
LIFETIME = Created during ALLOCATE processing.
Deleted during FREE processing.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDCD which identifies the
conversation which currently owns the connection.
Also located from DFHSZDCM on two chains:
(1) All active conversations.
(2) All inactive conversations. ie. those
conversations relinquished with FREE(PASS).
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	116	DFHSZDCV	
(0)	CHARACTER	32	SZD_CV_EYE	eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
	These fields chain the conversation off of DFHSZDCM. A conversation exists on one or other of the inactive or active conversation lists.			
(20)	ADDRESS	4	SZD_CV_PREV	previous conversation
(24)	ADDRESS	4	SZD_CV_NEXT	next conversation
	Associated connection			
(28)	ADDRESS	4	SZD_CV_CDPTR	connection address
	Maximum buffer size allowed on conversation.			
(2C)	FULLWORD	4	SZD_CV_BSIZE	
(2C)	ADDRESS	4	SZD_CV_PDPTR	browse pool

Offset Hex	Type	Len	Name (Dim)	Description
(2C)	ADDRESS	4	SZD_CV_PSPTR	browse property
Conversation ID. Constructed during ALLOCATE processing. It uniquely identifies a particular conversation.				
(30)	BITSTRING	8	SZD_CV_ID	
(30)	ADDRESS	4	SZD_CV_NDPTR	browse node
(30)	ADDRESS	4	SZD_CV_IDX	
(34)	ADDRESS	4	SZD_CV_TDPTR	browse target
(34)	ADDRESS	4	SZD_CV_IDY	
The following three fields combine to uniquely identify the present owner of the conversation. When a conversation is inactive then these are zero.				
(38)	CHARACTER	12	SZD_CV_TID	collective terminal ID
(38)	CHARACTER	4	SZD_CV_TRANID	
(3C)	CHARACTER	4	SZD_CV_TERMID	
(40)	CHARACTER	4	SZD_CV_TASK_NUM	
This field is the root for a list of API requests scheduled for this conversation.				
(44)	ADDRESS	4	SZD_CV_APIQ	
(44)	HALFWORD	2	SZD_CV_RTYPE	BROWSE request type
(46)	HALFWORD	2	*	padding
Conversation control flags				
(48)	BITSTRING	4	SZD_CV_FLAGS	
	1...		SZD_CV_BROWSE	This is a BROWSE conversation
This corresponds to the unit-of-work identifier. It is presently unused.				
(4C)	CHARACTER	27	SZD_CV_FQCC	
(67)	CHARACTER	1	*	
(68)	FULLWORD	4	SZD_CV_BTSIZE	
(6C)	FULLWORD	4	SZD_CV_ECOUNT	
(70)	ADDRESS	4	SZD_CV_BTPTR	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	116	DFHSZDCV_LEN	

FEP08 Device support extension

CONTROL BLOCK NAME = DFHSZDDS
DESCRIPTIVE NAME = CICS (FEPI) Device Support Extension
FUNCTION = Contains device specific information associated
with a particular connection. 1 DFHSZDDS exists
for each defined DFHSZDCD within a pool designated
as being in formatted mode.
LIFETIME = Created by DFHSZRIC during INSTALL processing.
Deleted by DFHSZRDC during DISCARD processing.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDCD which describes the
connection to which this extension relates. The
DCD may be located from the DFHSZDPD which owns
the connection.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	248	DFHSZDDS	
(0)	CHARACTER	32	SZD_DS_EYE	eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	ADDRESS	4	SZD_DS_PREV	previous element
(24)	ADDRESS	4	SZD_DS_NEXT	next element
(28)	FULLWORD	4	SZD_DS_TYPE	next element
(2C)	BITSTRING	4	SZD_DS_FLAGS	next element

End of portion that must match DFHSZDQE
P1GPTR is also the base address of the area whose length
is contained in DLENGTH. This is the address used to release
storage if the connection is discarded.
P1APTR thru P1CPTR are the base addresses of the various
attribute planes needed to support 3270. The storage for all
of the planes is obtained at BIND time.
P1CPTR is only allocated if one of the 3279 device-types
was specified.
P1X, P1S and P1V are only allocated if the EDS flag is set
in the LU profile at BIND time.
This allows for a storage efficient operating mode of non-EDS
monochrome.

(30)	ADDRESS	4	SZD_DS_P1GPTR	graphic plane pointer
(34)	ADDRESS	4	SZD_DS_P1APTR	attribute plane
(38)	ADDRESS	4	SZD_DS_P1XPTR	ext. hilite plane
(3C)	ADDRESS	4	SZD_DS_P1SPTR	Char. selection plane
(40)	ADDRESS	4	SZD_DS_P1VPTR	xparency/validation
(44)	ADDRESS	4	SZD_DS_P1CPTR	Colour plane

CCP is the current cursor position. It is affected by inbound
datastream and by API keystroke or image data.

(48)	FULLWORD	4	SZD_DS_CCP	current cursor pos.

CBA provides a common index value into all of the data planes
identified above. It represents the 3270's perception of
where buffer activity will take place.

(4C)	FULLWORD	4	SZD_DS_CBA	current buffer address
(50)	FULLWORD	4	SZD_DS_TBA	temp. buffer address
(54)	FULLWORD	4	SZD_DS_DBA	dest. buffer address
(58)	FULLWORD	4	SZD_DS_SENSE	last sense code
(5C)	ADDRESS	4	SZD_DS_CDPTR	connection address
(60)	FULLWORD	4	SZD_DS_DLENGTH	dynamic area size
(64)	FULLWORD	4	SZD_DS_KINDEX	keystroke bfr index
(68)	FULLWORD	4	SZD_DS_LA	last attribute index
(6C)	FULLWORD	4	SZD_DS_IDPTR	input data index
(70)	FULLWORD	4	SZD_DS_MDPTR	modified data index
(74)	ADDRESS	4	SZD_DS_IDATA	input data address
(78)	FULLWORD	4	SZD_DS_IDLEN	input data length
(7C)	FULLWORD	4	SZD_DS_CHAIN	chain save area

Offset Hex	Type	Len	Name (Dim)	Description

Implicit partition (00) dimension information
PSIZE is calculated at BIND time and is used to determine the amount of dynamic storage required and to detect wraparound during buffer processing. It is recalculated each time the session is bound or an ERASE/WRITE is received. Default default and alternate sizes are set based upon the device-type value provided in the pool. When the BIND is received, the BIND values override. The BIND also determines whether or not the device can switch between default and alternate.

Offset Hex	Type	Len	Name (Dim)	Description
(80)	FULLWORD	4	SZD_DS_PSIZE	plane size
(84)	BITSTRING	1	SZD_DS_PSX	PS width (current)
(85)	BITSTRING	1	SZD_DS_PSY	PS depth -do-
(86)	BITSTRING	1	SZD_DS_PSXDEF	PS width (default)
(87)	BITSTRING	1	SZD_DS_PSYDEF	PS depth -do-
(88)	BITSTRING	1	SZD_DS_PSXALT	PS width (alternate)
(89)	BITSTRING	1	SZD_DS_PSYALT	PS depth -do-
(8A)	BITSTRING	1	*	reserved not available
(8B)	BITSTRING	1	*	reserved -do-

Note that the following byte is reset to zero whenever a BIND is processed.

Offset Hex	Type	Len	Name (Dim)	Description
(8C)	BITSTRING	1	SZD_DS_CONTROL	PS control flags
	1...		SZD_DS_GATE	API queue gate flag
	.1..		SZD_DS_INOP	inbound operation
	..1.		SZD_DS_TWAIT	input inhibit flag
	...1		SZD_DS_SLOCK	system lock
	.... 1...		SZD_DS_ALARM	alarm has sounded
	.... .1..		SZD_DS_KLOCK	keyboard is locked
	.... ..1.		SZD_DS_MDR	modified data ready
	1		SZD_DS_IFLAG	pending input
(8D)	BITSTRING	1	SZD_DS_FLAG3	more flags
	1...		SZD_DS_L1PROT	prot stat (loc(0))
	.1..		SZD_DS_CPPROT	prot stat (CCP)
	..1.		SZD_DS_AFLAG	formatted flag
	...1		SZD_DS_INS	insert flag
	.... 1...		SZD_DS_POST	SEND POST memory
	.... .1..		SZD_DS_RMT	attention type
	.... ..1.		SZD_DS_PBB	Pending begin-bracket
	1		SZD_DS_PSI	PSpace invalid

Datastream sequencing control flags. Due to the nature of buffering, the 3270 can never assume that all of the bytes associated with an attribute, order or structured field are present, it must assume that each byte could be its last. These flags are used to monitor the present condition of the outbound datastream.

Offset Hex	Type	Len	Name (Dim)	Description
(8E)	BITSTRING	1	SZD_DS_SEQ1	PS control flags
	1...		SZD_DS_SB	SBA order received
	.1..		SZD_DS_SA	SA order received
	..1.		SZD_DS_RA	RA detected
	...1		SZD_DS_GE	graphic escape detect
	.... 1...		SZD_DS_SF	SF order received
	.... .1..		SZD_DS_EU	EUA order received
	.... ..1.		SZD_DS_MF	modify field
	1		SZD_DS_SE	Start field extended
(8F)	BITSTRING	1	SZD_DS_SEQ2	
	1...		SZD_DS_RA1	RA 1st byte
	.1..		SZD_DS_SB1	SBA 1st address stored
	..1.		SZD_DS_RA2	RA 2nd byte
	...1		SZD_DS_CMD	cmd/order processed
	.... 1...		SZD_DS_EU1	EUA addr byte 1 stored
	.... .1..		SZD_DS_SENDREQ	SEND requested
	.... ..1.		SZD_DS_WSFREQ	Query Reply required
	1		SZD_DS_WSFIP	WSF in progress

CC is the 3270 IO command code currently being processed, ie WRITE, READ MODIFIED etc.
WC is the currently-in-effect WCC byte. For outbound 3270DS structured fields, these values may change several times within a single transmission.
AID is the last inbound attention-identifier. This is reset when activity causes the 3270 to exit the inbound-pending state.
Currently, a PID of 00 is mandatory. The BFLAG field is the capability byte of the LU profile (from the BIND). It is stored at OPNSEC time.

Offset Hex	Type	Len	Name (Dim)	Description
(90)	BITSTRING	1	SZD_DS_CC	last IO command code
(91)	BITSTRING	1	SZD_DS_WC	last write control
	1...		*	reserved
	.1..		SZD_DS_WC_RESET	reset control
	..1.		SZD_DS_WC_P1	printer
	...1		SZD_DS_WC_P2	control
	.... 1...		SZD_DS_WC_SP	start print
	.... .1..		SZD_DS_WC_ALARM	sound the alarm

Offset Hex	Type	Len	Name (Dim)	Description
	.... ..1.		SZD_DS_WC_KENA	enable the keyboard
	1		SZD_DS_WC_RMDT	reset MDT flags
(92)	BITSTRING	1	SZD_DS_AID	current attention ID
(93)	BITSTRING	1	SZD_DS_INPID	inbound partition ##
(94)	BITSTRING	1	SZD_DS_CCBYTE	current colour info
	1111		SZD_DS_CBG	
	 1111		SZD_DS_CFG	
(95)	BITSTRING	1	SZD_DS_CXBYTE	current ext, highlight
	1111		SZD_DS_CXP	
	 1111		SZD_DS_CXA	
(96)	BITSTRING	1	SZD_DS_CSBYTE	current characer set
(97)	BITSTRING	1	SZD_DS_CVBYTE	current validation
	1111		SZD_DS_CFV	
	 1111		SZD_DS_CFO	
(98)	BITSTRING	1	SZD_DS_DCBYTE	default colour info
	1111		SZD_DS_DBG	
	 1111		SZD_DS_DFG	
(99)	BITSTRING	1	SZD_DS_DXBYTE	default ext, highlight
	1111		SZD_DS_DXP	
	 1111		SZD_DS_DXA	
(9A)	BITSTRING	1	SZD_DS_DSBYTE	default characer set
(9B)	BITSTRING	1	SZD_DS_DVBYTE	default validation
	1111		SZD_DS_DFV	
	 1111		SZD_DS_DFO	
(9C)	BITSTRING	1	SZD_DS_ATLIM	max PA count
(9D)	BITSTRING	1	SZD_DS_PFLIM	max PF count
(9E)	BITSTRING	1	SZD_DS_DABYTE	default attribute
(9F)	BITSTRING	1	SZD_DS_WSFCC	SF command byte
Device level control information				
(A0)	BITSTRING	1	SZD_DS_BFLAG	BIND EDS byte
	1...		SZD_DS_EDS	EDS indicator
	.1..		SZD_DS_NFIP	NULL fill in progress
(A1)	BITSTRING	1	SZD_DS_SAT	SA order attrib. type
(A2)	BITSTRING	2	SZD_DS_SFLEN	structured field length
(A2)	BITSTRING	1	SZD_DS_SFLEN1	structured field length
(A3)	BITSTRING	1	SZD_DS_SFLEN2	
(A4)	BITSTRING	1	SZD_DS_SFID	SF id byte
(A5)	BITSTRING	1	SZD_DS_SFID2	second structure ID
(A6)	BITSTRING	1	SZD_DS_SFPID	partition ID
(A7)	BITSTRING	1	SZD_DS_SFTYPE	SF type byte
Device related SF data area				
(A8)	CHARACTER	68	SZD_DS_SFDATA	structured field info
(A8)	BITSTRING	2	SZD_DS_QLEN	QUERY REPLY length
(AA)	BITSTRING	1	SZD_DS_QID	QUERY REPLY ID byte
(AB)	BITSTRING	1	SZD_DS_QCODE	QUERY REPLY code byte
(AC)	AREA	64	SZD_DS_QDATA	QUERY REPLY data area
(AC)	BITSTRING	1	SZD_DS_TB1	temp. buffer address
(AD)	BITSTRING	1	SZD_DS_TB2	temp. buffer address
(AE)	BITSTRING	1	SZD_DS_SEC	attribute counter
(AF)	BITSTRING	1	SZD_DS_SET	attribute type
(EC)	BITSTRING	1	SZD_DS_DFLAGS	Device flags
	1...		SZD_DS_COLOUR	colour is supported
	.1..		SZD_DS_TPS	TPS device
	..1.		SZD_DS_SFL1	SF length byte flag
	...1		SZD_DS_SFL2	SF length byte flag
	.... 1...		SZD_DS_DFLEN	Default length flag
	.... .1..		SZD_DS_RIP	RECEIVE in progress
	.... ..1.		SZD_DS_ERI	Erase required
	1		SZD_DS_MSIP	mag stripe current
(ED)	BITSTRING	1	SZD_DS_QP_FLAG1	Query partition flags
	1...		SZD_DS_QP_CHARS	Character sets
	.1..		SZD_DS_QP_ASIA	DBCS Asia
	..1.		SZD_DS_QP_IMPA	Implicit partition
	...1		SZD_DS_QP_USEA	Usable area
	.... 1...		SZD_DS_QP_SUMM	Summary
(EE)	BITSTRING	1	SZD_DS_QP_FLAG2	Query partition flags
	1...		SZD_DS_QP_TRAN	Transparency
	.1..		SZD_DS_QP_ALPHA	Alphanumeric part.
	..1.		SZD_DS_QP_COLOR	Color
	...1		SZD_DS_QP_OUTL	Outlining
	.... 1...		SZD_DS_QP_VALI	Validation
	.... .1..		SZD_DS_QP_HILI	Highlighting
(EF)	BITSTRING	1	*	
	1...		SZD_DS_SFPIDX	PID memory flag
(F0)	FULLWORD	4	SZD_DS_RDPTR	Received data index
(F4)	FULLWORD	4	SZD_DS_END	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1	ABYTE	field attribute byte
	1...		*	
	.1..		*	

Offset Hex	Type	Len	Name (Dim)	Description
	..1.		SZD_DS_PROT	protected field flag
	...1		SZD_DS_NUM	alphanumeric flag
	.... 1...		SZD_DS_DS1	display/selector pen
	.... .1..		SZD_DS_DS2	control bits
	.... ..1.		*	
	1		SZD_DS_MDT	modified data tag

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1	CBYTE	colour select buffer
	1111		SZD_DS_BG	background
	 1111		SZD_DS_FG	foreground

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1	XBYTE	extended highlighting
	1111		SZD_DS_XP	transparency control
	 1111		SZD_DS_XA	highlight value mask

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1	VBYTE	validation/outlining
	1111		SZD_DS_FV	validation mask
	 1111		SZD_DS_FO	outline mask

Constants

Len	Type	Value	Name	Description
4	DECIMAL	248	DFHSZDDS_LEN	

FEP09 Tsf - eye catcher map

CONTROL BLOCK NAME = DFHSZDEC
DESCRIPTIVE NAME = CICS (TSF) Eye Catcher Map
FUNCTION = Provides mapping for the TSF data area
    eye-catcher.
LIFETIME = N/A. The eyecatcher is part of all other TSF
    data structures.
STORAGE CLASS = 31-bit addressable.
LOCATION = N/A. The eyecatcher is part of all other TSF
    data structures.
INNER CONTROL BLOCKS =
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS =
 MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
  DATA AREAS =
  CONTROL BLOCKS =
  GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	DFHSZDEC	
(0)	HALFWORD	2	SZD_EC_LENGTH	AREA LENGTH INCLUDING EC
(2)	CHARACTER	1	SZD_EC_GT	"GREATER-THAN" SIGN
(3)	CHARACTER	8	SZD_EC_NAME	DATA AREA NAME
(B)	CHARACTER	5	*	PADDING
(10)	CHARACTER	8	SZD_EC_SPID	SUBPOOL TOKEN
(18)	UNSIGNED	4	SZD_EC_CBID	PADDING
(1C)	CHARACTER	4	*	PADDING

Constants

Len	Type	Value	Name	Description
4	DECIMAL	32	DFHSZDEC_LEN	

FEP10 Node descriptor

CONTROL BLOCK NAME = DFHSZDND
DESCRIPTIVE NAME = CICS (FEPI) Node descriptor
FUNCTION = Contains the information needed by the resource
manager to control and support a front-end
node. A node exists for each VTAM ACB used by
the resource manager to communicate with the
network.
LIFETIME = Created by DFHSZRIN during INSTALL processing.
Deleted by DFHSZRDN during DISCARD processing.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDCM.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	212	DFHSZDND	
(0)	CHARACTER	32	SZD_ND_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	CHARACTER	24	SZD_ND_WE	ND WE
(20)	BITSTRING	8	SZD_ND_QCB	ND QCB
(20)	ADDRESS	4	SZD_ND_QP	Previous element
(24)	ADDRESS	4	SZD_ND_QC	Next element
(28)	FULLWORD	4	SZD_ND_REQ	Request type
(2C)	BITSTRING	4	*	unused
	1...		*	reserved - not available
	.1..		*	reserved - not available
	..1.		SZD_ND_ON_Q	On the process Q
	...1		SZD_ND_ON_QIRB	On the IRB process Q
	.... 1...		SZD_ND_ON_TMR	On the timer queue
	.... .1..		*	Reserved - not available
	.... ..1.		SZD_ND_ ON_QTPEND8	On the TPEND code 8 proc. Q

NOTE
End of section that must match DFHSZDQE

(30)	HALFWORD	2	SZD_ND_TRINTVL	Timer retry interval
(32)	HALFWORD	2	SZD_ND_TRTYPE	Timer retry type
(34)	CHARACTER	4	SZD_ND_DEFTRAN	Saved transid @BA65235C

Binds received from unKnown partners are queued here by IRB routines. Each entry is mapped by DFHSZDBI.

(38)	BITSTRING	8	SZD_ND_BI_QCB	Node SZDBI list
(38)	ADDRESS	4	SZD_ND_BI_QC	DBI list header
(3C)	ADDRESS	4	*	unused - available
(40)	BITSTRING	4	SZD_ND_FLAGS	

Byte 0

	1...		SZD_ND_RECANYR	Receive Any Queued
	.1..		SZD_ND_RECANYN	Receive Any Needed
	..1.		SZD_ND_SLFAIL	SETLOGON failed
	...1		SZD_ND_SLMEM	SETLOGON could not be issue buffer not available
	.... 1...		SZD_ND_TPEND_0	TPEND scheduled with code 0
	.... .1..		SZD_ND_TPEND_4	TPEND scheduled with code 4
	.... ..1.		SZD_ND_TPEND_8	TPEND scheduled with code 8
	1		SZD_ND_TPEND	TPEND scheduled

Byte 1

(41)	1...		SZD_ND_SHUT	SHUTDOWN initiated
	.1..		SZD_ND_CLOSE	close requested
	..1.		SZD_ND_DISCARD	DISCARD initiated
	...1		SZD_ND_IMMED	unconditional closure
	.... 1...		SZD_ND_OPENREQ	OPEN requested
	.... .1..		SZD_ND_OPENRIP	OPEN in progress
	.... ..1.		SZD_ND_OPENOK	OPENed OK
	1		SZD_ND_OPENFAIL	OPEN failed

Offset Hex	Type	Len	Name (Dim)	Description
Byte 2				
(42)	1...		SZD_ND_UNSOL	Unsolicited BIND received
	.1..		SZD_ND_UNSOLEX	BIND expected
	..1.		*	unused - available
	...1		*	unused - available
	.... 1...		*	unused - available
	.... .1..		*	unused - available
	.... ..1.		*	unused - available
	1		*	unused - available
Byte 3				
(43)	1...		SZD_ND_SLDONE	setlogon footprint
	.1..		SZD_ND_RADONE	receive any footprint
	..1.		*	unused - available
	...1		*	unused - available
	.... 1...		*	unused - available
	.... .1..		*	unused - available
	.... ..1.		*	unused - available
	1		*	unused - available
The address of a DRA is stored here whenever the RECEIVE(ANY) is satisfied for this node.				
(44)	ADDRESS	4	SZD_ND_RECANY	Receive Any Ptr
This is the configuration management portion of the data area, information kept here allows FEPI to define and delete the resource.				
(48)	CHARACTER	140	SZD_ND_API	
(48)	ADDRESS	4	SZD_ND_PREV	Prior DND
(4C)	ADDRESS	4	SZD_ND_NEXT	Next DND on list
(50)	ADDRESS	4	SZD_ND_CDLIST	connection list
(54)	ADDRESS	4	SZD_ND_SRLIST	surrogate list
(58)	ADDRESS	4	SZD_ND_ACB	associated ACB
(5C)	ADDRESS	4	SZD_ND_CM	common area ptr
(60)	ADDRESS	4	SZD_ND_ACPTR	ACB work area
(64)	CHARACTER	12	*	VTAM ACB name
(64)	CHARACTER	1	SZD_ND_NAMEL	
(65)	CHARACTER	8	SZD_ND_NAME	
(6D)	CHARACTER	3	*	
(70)	CHARACTER	12	*	ACB Password
(70)	CHARACTER	1	SZD_ND_PASSL	
(71)	CHARACTER	8	SZD_ND_ PASSWORD	
(79)	CHARACTER	3	*	
(7C)	HALFWORD	2	SZD_ND_ SERVSTATUS	service status
(7E)	HALFWORD	2	SZD_ND_ ACQSTATUS	actual network status
(80)	HALFWORD	2	SZD_ND_ DESSTATUS	desired network status
(82)	HALFWORD	2	SZD_ND_ INSTSTATUS	installation status
(84)	HALFWORD	2	SZD_ND_ASTAT	acb status
(86)	HALFWORD	2	SZD_ND_ERFLG	acb open failure code
(88)	ADDRESS	4	SZD_ND_CDSTQ	CLSDST connection queue
(8C)	FULLWORD	4	SZD_ND_USAGE	usage counter
(90)	FULLWORD	4	SZD_ND_RCOUNT	maximum open retries
(94)	CHARACTER	64	SZD_ND_UDATA	user data storage

Constants

Len	Type	Value	Name	Description
4	DECIMAL	212	DFHSZDND_LEN	

FEP11 Pool descriptor

CONTROL BLOCK NAME = DFHSZDPD
DESCRIPTIVE NAME = CICS (FEPI) Pool descriptor
FUNCTION = Acts as a correlator for connection, nodes and
targets. 1 DFHSZDPD exists for each pool defined
by the installation during INSTALL processing.
LIFETIME = Created by DFHSZRIP during INSTALL processing.
Deleted by DFHSZRDP during DISCARD processing.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDCM.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	316	DFHSZDPD	
(0)	CHARACTER	32	SZD_PD_EYE	eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	

This area chains the pool from DFHSZDCM. It is the list of pools known to the resource manager.

Offset Hex	Type	Len	Name (Dim)	Description
(20)	ADDRESS	4	SZD_PD_PREV	prev pool
(24)	ADDRESS	4	SZD_PD_NEXT	next pool
(28)	CHARACTER	8	SZD_PD_NAME	Pool name
(30)	CHARACTER	8	SZD_PD_PROPERTY	Propertset name

These lists identify the resources associated with the pool by configuration processing.

Offset Hex	Type	Len	Name (Dim)	Description
(38)	ADDRESS	4	SZD_PD_NDLIST	assoc. nodes
(3C)	ADDRESS	4	SZD_PD_TDLIST	assoc. Targets
(40)	ADDRESS	4	SZD_PD_CDLIST	assoc. conns.
(44)	ADDRESS	4	SZD_PD_AWLIST	q'd allocates
(48)	HALFWORD	2	SZD_PD_ SERVSTATUS	Pool service status
(4A)	HALFWORD	2	SZD_PD_ INSTSTATUS	Pool install status

This area is initialised from the contents of the property set named above. The values are copied at the time the association is made. The pool is not subsequently dependent upon the existence of the property-set.

Offset Hex	Type	Len	Name (Dim)	Description
(4C)	CHARACTER	132	SZD_PD_PROPS	Property values
(4C)	BITSTRING	2	SZD_PS_FLAGS	
(4C)	BITSTRING	1	*	
(4D)	1...		SZD_PS_EXCEPTIONQ_X	
	.1..		*	
	..1.		*	
	...1		*	
	.... 1...		SZD_PS_ENDSESSION_X	
	.... .1..		SZD_PS_ UNSOLDATA_X	
	.... ..1.		SZD_PS_BEGINSESSION_X	
	1		SZD_PS_STSN_X	
(4E)	BITSTRING	2	*	
(50)	ADDRESS	4	SZD_PS_ ENDSESSION	
(54)	ADDRESS	4	*	
(58)	FULLWORD	4	*	
(5C)	CHARACTER	4	SZD_PS_DEFTRAN	
(60)	FULLWORD	4	SZD_PS_ MAXFLENGTH	
(64)	CHARACTER	8	SZD_PS_ FJOURNALNAME	
(6C)	HALFWORD	2	SZD_PS_DEVICE	
(6E)	HALFWORD	2	SZD_PS_FORMAT	
(70)	HALFWORD	2	SZD_PS_ CONTENTION	
(72)	HALFWORD	2	SZD_PS_ INITIALDATA	
(74)	HALFWORD	2	SZD_PS_ UNSOLDATACK	
(76)	HALFWORD	2	SZD_PS_MSGJRNL	
(78)	CHARACTER	4	SZD_PS_STSN	
(7C)	CHARACTER	4	SZD_PS_ BEGINSESSION	

Offset Hex	Type	Len	Name (Dim)	Description
(80)	CHARACTER	4	SZD_PS_ UNSOLDATA	
(84)	CHARACTER	4	SZD_PS_ EXCEPTIONQ	
(88)	CHARACTER	8	*	
(90)	CHARACTER	64	SZD_PS_UDATA	
(D0)	CHARACTER	64	SZD_PD_UDATA	User data
Statistics counters				
(110)	FULLWORD	4	SZD_PD_TARGETS	# targets in pool *
(114)	FULLWORD	4	SZD_PD_NODES	# nodes in pool *
(118)	FULLWORD	4	SZD_PD_ CONNECTIONS	# connections
(11C)	FULLWORD	4	SZD_PD_ PKCONNECTIONS	peak # connections *
(120)	FULLWORD	4	SZD_PD_ALLOCATED	# conversations * currently allocated *
(124)	FULLWORD	4	SZD_PD_ PKALLOCATED	peak # concurrent allocates
(128)	FULLWORD	4	SZD_PD_ TOTALLOCATES	Total # conversation allocates
(12C)	FULLWORD	4	SZD_PD_ ALLOCATESWAITING	Current # allocates waiting
(130)	FULLWORD	4	SZD_PD_ PKALLOCATESWAITING	Peak # allocates waiting
(134)	FULLWORD	4	SZD_PD_ TOTALLOCATEWAITS	Total # allocates waited
(138)	FULLWORD	4	SZD_PD_TIMEOUTS	# allocates that timed out

Constants

Len	Type	Value	Name	Description
4	DECIMAL	316	DFHSZDPD_LEN	

FEP12 Properties list

CONTROL BLOCK NAME = DFHSZDPP
DESCRIPTIVE NAME = CICS (FEPI) - Properties List
FUNCTION = API Propertyset definition parameter list extension.
LIFETIME = Duration of the INSTALL request to which it relates.
STORAGE CLASS = 31-bit addressable.
LOCATION = Pointed to by DFHSZDRP.
INNER CONTROL BLOCKS =
NOTES :
 Dependencies = S/370
 Restrictions =
 Module type = Control block definition
EXTERNAL REFERENCES =
  Data areas =
  Control blocks =
  Global variables (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	104	DFHSZDPP	
(0)	CHARACTER	32	SZD_PP_EYE	Eye-catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	BITSTRING	2	SZD_PP_FLAGS	Features flags:
(20)	BITSTRING	1	*	*reserved*
(21)	1...		SZD_PP_ EXCEPTIONQ_X	- exceptional event Q
	.111		*	*reserved*
	.... 1...		SZD_PP_ ENDSESSION_X	- end-session tran
	.... .1..		SZD_PP_ UNSOLDATA_X	- unsol data tran
	.... ..1.		SZD_PP_ BEGINSESSION_X	- begin-session tran
	1		SZD_PP_STSN_X	- STSN tran

Offset Hex	Type	Len	Name (Dim)	Description
(22)	BITSTRING	2	*	*reserved*
(24)	HALFWORD	2	SZD_PP_DEVICE	Device
(26)	HALFWORD	2	SZD_PP_FORMAT	Data format
(28)	HALFWORD	2	SZD_PP_ CONTENTION	Contention
(2A)	HALFWORD	2	SZD_PP_ INITIALDATA	Initial inbound data
(2C)	HALFWORD	2	SZD_PP_MSGJRNL	Journal control
(2E)	HALFWORD	2	SZD_PP_ UNSOLDATACK	Unsol data response
(30)	CHARACTER	16	*	*reserved*
(40)	FULLWORD	4	SZD_PP_ MAXFLENGTH	Maximum data length
(44)	CHARACTER	4	SZD_PP_STSN	STSN tran
(48)	CHARACTER	4	SZD_PP_ BEGINSESSION	Begin-session tran
(4C)	CHARACTER	4	SZD_PP_UNSOLDATA	Unsolicited data tran
(50)	CHARACTER	4	SZD_PP_ EXCEPTIONQ	Exceptional event Q
(54)	CHARACTER	4	SZD_PP_ ENDSESSION	End -session tran
(58)	CHARACTER	4	*	*reserved*
(5C)	FULLWORD	4	SZD_PP_ FJOURNALNUM	Journal number
(60)	CHARACTER	8	SZD_PP_ FJOURNALNAME	Journal name
(68)	CHARACTER		*	End of property list

Constants

Len	Type	Value	Name	Description
4	DECIMAL	104	DFHSZDPP_LEN	

FEP13 Property set info

CONTROL BLOCK NAME = DFHSZDPS
DESCRIPTIVE NAME = CICS (FEPI) Property Set information
FUNCTION = Describes the functional properties for a pool
of resources with which the set is related.
1 control block will exist for each unique set
of characteristics defined by the installation
during INSTALL processing.
LIFETIME = Created by DFHSZRIS during INSTALL processing.
Deleted by DFHSZRDS during DISCARD processing.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDCM.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	180	DFHSZDPS	
(0)	CHARACTER	32	SZD_PS_EYE	eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	

This area chains the property-set of DFHSZDCM. This is the list of property-sets known to the resource manager.

Offset Hex	Type	Len	Name (Dim)	Description
(20)	ADDRESS	4	SZD_PS_PREV	previous propertyset
(24)	ADDRESS	4	SZD_PS_NEXT	next property set
(28)	CHARACTER	8	SZD_PS_NAME	name of this prop. set

The following fields contain the information the constitutes a property-set.
It is copied to the DFHSZDPD whenever a pool is defined and associated with a property-set.

Offset Hex	Type	Len	Name (Dim)	Description
(30)	CHARACTER	132	SZD_PS_PROPS	
(30)	BITSTRING	2	SZD_PS_FLAGS	profile flags
(30)	BITSTRING	1	*	

Offset Hex	Type	Len	Name (Dim)	Description
(31)	1...		SZD_PS_EXCEPTIONQ_X	
	.1..		*	
	..1.		*	
	...1		*	
	.... 1...		SZD_PS_ENDSESSION_X	
	.... .1..		SZD_PS_ UNSOLDATA_X	
	.... ..1.		SZD_PS_BEGINSESSION_X	
	1		SZD_PS_STSN_X	
(32)	BITSTRING	2	*	reserved - not available
(34)	ADDRESS	4	SZD_PS_ ENDSESSION	FREE transaction
(38)	ADDRESS	4	*	reserved
(3C)	FULLWORD	4	*	reserved
(40)	CHARACTER	4	SZD_PS_DEFTRAN	Saved Tranid @BA65235C
(44)	FULLWORD	4	SZD_PS_ MAXFLENGTH	max data size allowed
(48)	CHARACTER	8	SZD_PS_ FJOURNALNAME	msg journal name
(50)	HALFWORD	2	SZD_PS_DEVICE	device type emulated
(52)	HALFWORD	2	SZD_PS_FORMAT	datastream/bufferd
(54)	HALFWORD	2	SZD_PS_ CONTENTION	contention rules
(56)	HALFWORD	2	SZD_PS_ INITIALDATA	Rule for init. data
(58)	HALFWORD	2	SZD_PS_ UNSOLDATACK	Rule for unsol. data
(5A)	HALFWORD	2	SZD_PS_MSGJRNL	Message journalling
(5C)	CHARACTER	4	SZD_PS_STSN	STSN transaction
(60)	CHARACTER	4	SZD_PS_ BEGINSESSION	Init. data xaction
(64)	CHARACTER	4	SZD_PS_ UNSOLDATA	Unsolicited data xaction
(68)	CHARACTER	4	SZD_PS_ EXCEPTIONQ	Exception event TD q
(6C)	CHARACTER	8	*	*reserved*
(74)	CHARACTER	64	SZD_PS_UDATA	user data

Constants

Len	Type	Value	Name	Description
4	DECIMAL	180	DFHSZDPS_LEN	

FEP14 Work queue element

CONTROL BLOCK NAME = DFHSZDQE
DESCRIPTIVE NAME = CICS (FEPI) Work queue element
FUNCTION = Represents and correlates processing to be
performed on behalf of a front-end application
program. 1 block will exist for each current
work request.
LIFETIME = Created by DFHSZRPW during adaptor request
preparation. Deleted by DFHSZRRT during adaptor
request cleanup.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDCM.
INNER CONTROL BLOCKS =
NOTES :
Dependencies = S/370
Restrictions =
Module type = Control block definition
EXTERNAL REFERENCES =
Data areas =
Control blocks = DFHSZDEC (Eyecatcher structure definition)
Global variables (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	332	DFHSZDQE	
(0)	CHARACTER	40	SZD_QE_PREFIX	RM private prefix
(0)	CHARACTER	32	SZD_QE_EYE	Eye-catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	ADDRESS	4	SZD_QE_PREV	previous dqe in queue
(24)	ADDRESS	4	SZD_QE_NEXT	next dqe in the queue
Start of public area. This is the section of the DQE updated by the adaptor during request initialisation.				
(28)	CHARACTER	68	SZD_QE_PUBLIC	External area
(28)	FULLWORD	4	SZD_QE_REQTYPE	Request type
(2C)	BITSTRING	1	SZD_QE_REQFLAG	Request flags:
	1...		SZD_QE_REQFLAG_POST	- POST needed
	.1..		SZD_QE_EXPFLAG	- expedited
	..1.		SZD_QE_ON_PRB	Queued by PRB
	...1		SZD_QE_ON_IRB	Queued by IRB
	.... 1...		SZD_QE_ON_TMR	Queued by TMR
	.... .1..		SZD_QE_ON_API	Queued by API
	.... ..1.		SZD_QE_ON_TP8	Queued by TPEND code 8
	1		SZD_QE_POSTED	Request completed
Timer Services Control Bits				
(2D)	BITSTRING	1	*	Timer Services Flags
	1...		SZD_QE_TIMED	Request requires timing
	.1..		SZD_QE_ TIMED_OUT	Request abandoned
	..1.		SZD_QE_PURGE	RM must free element
	...1		SZD_QE_ RRT_SEEN	Owner has exited flag
(2E)	BITSTRING	1	*	Unused available
(2F)	BITSTRING	1	*	Misc flags @BA66310C
	1...		SZD_QE_ CONFDATA	CONFDATA=YES @BA66310A
(30)	ADDRESS	4	SZD_QE_REQDATA	Request area address
(34)	ADDRESS	4	SZD_QE_CHAIN	Next dqe in chain pointer
(38)	CHARACTER	8	SZD_QE_CONVID	Conversation ID
(40)	BITSTRING	4	SZD_QE_ECB	CICS thread ECB
(44)	CHARACTER	27	SZD_QE_FQCC	FQCC
(5F)	CHARACTER	1	*	Padding
(60)	CHARACTER	12	SZD_QE_TID	Collective ID
(60)	CHARACTER	4	SZD_QE_TRANID	Transaction ID
(64)	CHARACTER	4	SZD_QE_TERMID	Terminal ID
(68)	CHARACTER	4	SZD_QE_TASKNUM	CICS task number
Start of resource manager private suffix				
(6C)	CHARACTER	224	SZD_QE_PRIVATE	Internal area
(6C)	ADDRESS	4	SZD_QE_DATA	Assoc. stg address
(70)	FULLWORD	4	SZD_QE_DATALEN	Assoc. stg length
(74)	ADDRESS	4	SZD_QE_CVPTR	Conversation address

Offset Hex	Type	Len	Name (Dim)	Description
Timer services area. TOCK contains the TICK value at which the request should be timed-out. TNEXT and TPREV chain time-out-able requests together. This chain is then scanned by timer services. The request is added to the timer-chain when the request is allocated by PW (if a timeout was requested). It is removed by DFHSZRRT of timer services.				
(78)	FULLWORD	4	SZD_QE_TOCK	Expiry time
(7C)	FULLWORD	4	SZD_QE_TICK	SOP time record
(80)	ADDRESS	4	SZD_QE_TPREV	Next DQE in timer Q
(84)	ADDRESS	4	SZD_QE_TNEXT	Next DQE in timer Q
(88)	ADDRESS	4	SZD_QE_TARGET	Chosen target fo alloc *
This MUST come last				
(8C)	AREA	192	SZD_QE_RP	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	332	DFHSZDQE_LEN	

FEP15 VTAM receive request block

CONTROL BLOCK NAME = DFHSZDRA
DESCRIPTIVE NAME = CICS (FEPI) VTAM Receive Request Block
FUNCTION =
Defines the VTAM Receive Requests Block.
This data area is a part of the FEPI Resource Manager.
It defines the format of the
VTAM Receive Request Block.
Lifetime = While a VTAM Receive request is active
Storage class = 31-bit addressable
Location = Chained from Common block
Inner control blocks = Not applicable
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Data Area
EXTERNAL REFERENCES = None
DATA AREAS = None
CONTROL BLOCKS = None
GLOBAL VARIABLES = None

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	184	DFHSZDRA	
(0)	CHARACTER	32	SZD_RA_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	BITSTRING	8	SZD_RA_QEB	
(20)	ADDRESS	4	*	unused available
(24)	ADDRESS	4	SZD_RA_QNEXT	Points to next in chain
(28)	CHARACTER	8	*	
(28)	FULLWORD	4	SZD_RA_REQTYPE	reserved
(2C)	BITSTRING	4	SZD_RA_FLAGS	reserved
(30)	HALFWORD	2	SZD_RA_TRINTVL	timer retry interval
(32)	HALFWORD	2	SZD_RA_TRTYPE	timer retry type
(34)	ADDRESS	4	SZD_RA_DYNAA	unused available
(38)	ADDRESS	4	SZD_RA_CM	common area ptr
(3C)	ADDRESS	4	SZD_RA_CD	connection ptr
(40)	ADDRESS	4	SZD_RA_ND	node area ptr
(44)	FULLWORD	4	SZD_RA_DYNAL	unused available
(48)	CHARACTER	112	SZD_RA_RPL	VTAM RPL
(48)	AREA	112	SZD_RA_VTAM	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	184	DFHSZDRA_LEN	

FEP16 VTAM requests block

CONTROL BLOCK NAME = DFHSZDRB
DESCRIPTIVE NAME = CICS (FEPI) VTAM Requests Block
FUNCTION =
  Defines the VTAM Requests Block.
  This data area is a part of the FEPI Resource Manager.
  It defines the format of the VTAM Requests Block.
Lifetime = While a VTAM request is active
Storage class = 31-bit addressable
Location = Chained from Common block
Inner control blocks = Not applicable
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS = None
 MODULE TYPE = Data Area
EXTERNAL REFERENCES = None
  DATA AREAS = None
  CONTROL BLOCKS = None
  GLOBAL VARIABLES = None

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	184	DFHSZDRB	
(0)	CHARACTER	32	SZD_RB_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	BITSTRING	8	SZD_RB_QEB	
(20)	ADDRESS	4	*	unused - available
(24)	ADDRESS	4	SZD_RB_QNEXT	Points to next in chain
(28)	CHARACTER	8	*	
(28)	FULLWORD	4	SZD_RB_REQTYPE	reserved
(2C)	BITSTRING	4	SZD_RB_FLAGS	reserved
(30)	HALFWORD	2	SZD_RB_TRINTVL	timer retry interval
(32)	HALFWORD	2	SZD_RB_TRTYPE	timer retry type
(34)	ADDRESS	4	SZD_RB_DYNAA	dynamic area pointer
(38)	ADDRESS	4	SZD_RB_CM	common area ptr
(3C)	ADDRESS	4	SZD_RB_CD	connection ptr
(40)	ADDRESS	4	SZD_RB_ND	node ptr
(44)	FULLWORD	4	SZD_RB_DYNAL	dynamic area length
(48)	CHARACTER	112	SZD_RB_RPL	VTAM RPL
(48)	AREA	112	SZD_RB_VTAM	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	184	DFHSZDRB_LEN	

FEP17 Request parameter area

CONTROL BLOCK NAME = DFHSZDRP
DESCRIPTIVE NAME = CICS (FEPI) - Request parameter area
FUNCTION = Contains the parameters associated with an
individual work request. One will exist for
each active processing request.
LIFETIME = Exists for the life of an API request.
STORAGE CLASS = 31-bit addressable.
LOCATION = Located from the DFHSZDQE to which the parameters
relate.
INNER CONTROL BLOCKS =
NOTES :
Dependencies = S/370
Restrictions =
Module type = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	192	DFHSZDRP	
(0)	CHARACTER	32	SZD_RPA_EYE	Eye-catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	CHARACTER	80	SZD_RIA	Request input area
(20)	HALFWORD	2	SZD_RIA_REQSUB	Request subtype
(22)	HALFWORD	2	*	*reserved*
(24)	FULLWORD	4	SZD_RIA_REQTYPE	Request type
(28)	FULLWORD	4	*	*reserved*
(2C)	BITSTRING	2	SZD_RIA_FLGS	Flags
(2C)	BITSTRING	1	*	*reserved*
(2D)	11..		*	*reserved*
	..1.		SZD_RIA_RU	RU
	...1		SZD_RIA_CHAIN	Chain
	...1		SZD_RIA_ BNEXTTARGET	Browse next target
	.... 1...		SZD_RIA_ ENDTASK	End of task
	.... 1...		SZD_RIA_FMH	FMH
	.... 1...		SZD_RIA_ BNEXTNODE	Browse next node
	.... .1..		SZD_RIA_PASS	Pass
	.... .1..		SZD_RIA_BEND	Browse end
	.... .1..		SZD_RIA_ CURSOR_X	Cursor set
	.... ..1.		SZD_RIA_ RELEASE	Release
	.... ..1.		SZD_RIA_ KEYSTROKES	Keystroke
	.... ..1.		SZD_RIA_ BNEXT	Browse next
	.... ..1.		SZD_RIA_ IMMEDIATE	Immediate
	1		SZD_RIA_FORCE	Force
	1		SZD_RIA_ INVITE	Invite
	1		SZD_RIA_ BSTART	Browse start
	1		SZD_RIA_ CONVERSE	Converse pool
	1		SZD_RIA_ LOCATION	Field by location
(2E)	BITSTRING	1	*	*reserved*
(2F)	BITSTRING	1	*	*reserved*
(30)	HALFWORD	2	SZD_RIA_OPT1	Option 1
(30)	HALFWORD	2	SZD_RIA_ CONTROL	Control
(30)	HALFWORD	2	SZD_RIA_ SERVSTATUS	Service status
(30)	CHARACTER	1	SZD_RIA_ RESET	Reset stats?

Offset Hex	Type	Len	Name (Dim)	Description
(31)	CHARACTER	1	SZD_RIA_ COLLECT	Collect stats?
(31)	CHARACTER	1	SZD_RIA_AID	AID
(31)	CHARACTER	1	SZD_RIA_ ESCAPE	Escape
(32)	HALFWORD	2	SZD_RIA_OPT2	Option 2
(32)	HALFWORD	2	SZD_RIA_VALUE	Value
(32)	HALFWORD	2	SZD_RIA_ ACQSTATUS	Acquire status
(32)	CHARACTER	1	*	*
(33)	CHARACTER	1	SZD_RIA_EOD	End of day stats? *
(34)	FULLWORD	4	*	*reserved*
(38)	FULLWORD	4	SZD_RIA_VAL1	Value 1
(38)	FULLWORD	4	SZD_RIA_ POOLNUM	Pool list count
(38)	FULLWORD	4	SZD_RIA_ TARGETNUM	Target list count
(38)	FULLWORD	4	SZD_RIA_ SENSEDATA	Sense data
(3C)	FULLWORD	4	SZD_RIA_VAL2	Value 2
(3C)	FULLWORD	4	SZD_RIA_ DATALEN	Data length
(3C)	FULLWORD	4	SZD_RIA_ MAXFLENGTH	Maximum length
(3C)	FULLWORD	4	SZD_RIA_ NODENUM	Node list count
(40)	FULLWORD	4	SZD_RIA_VAL3	Value 3
(40)	ADDRESS	4	SZD_RIA_LST3	List 3
(40)	ADDRESS	4	SZD_RIA_DATA	Data address
(40)	ADDRESS	4	SZD_RIA_ TARGETLIST	Target list
(40)	ADDRESS	4	SZD_RIA_ POOLLIST	Pool list
(40)	ADDRESS	4	SZD_RIA_ STATS	Stats buffer
(44)	FULLWORD	4	SZD_RIA_VAL4	Value 4
(44)	ADDRESS	4	SZD_RIA_LST4	List 4
(44)	FULLWORD	4	SZD_RIA_ FIELDNUM	Field number
(44)	FULLWORD	4	SZD_RIA_ FIELDLOC	Field location
(44)	ADDRESS	4	SZD_RIA_ NODELIST	Node list
(48)	FULLWORD	4	SZD_RIA_VAL5	Value 5
(48)	ADDRESS	4	SZD_RIA_LST5	List 5
(48)	FULLWORD	4	SZD_RIA_ TIMEOUT	Timeout
(48)	FULLWORD	4	SZD_RIA_ CURSOR	Cursor
(48)	ADDRESS	4	SZD_RIA_ PROPS	Properties data
(48)	ADDRESS	4	SZD_RIA_ APPLLIST	Appl names list
(48)	ADDRESS	4	SZD_RIA_ PASSWORDLIST	Password list
(4C)	FULLWORD	4	SZD_RIA_VAL6	Value 6
(4C)	ADDRESS	4	SZD_RIA_ USERDATA	User data address
(50)	CHARACTER	8	SZD_RIA_INC1	Inchar 1
(50)	CHARACTER	8	SZD_RIA_CONVID	Conv ID
(50)	CHARACTER	8	SZD_RIA_ PASSCONVID	Conv ID
(50)	CHARACTER	8	SZD_RIA_POOL	Pool
(58)	CHARACTER	8	SZD_RIA_INC2	Inchar 2
(58)	CHARACTER	8	SZD_RIA_TARGET	Target
(60)	CHARACTER	8	SZD_RIA_INC3	Inchar 3
(60)	CHARACTER	8	SZD_RIA_NODE	Node
(60)	CHARACTER	8	SZD_RIA_ PROPERTYSET	Property set
(60)	CHARACTER	4	SZD_RIA_ TRANSID	Transaction ID
(64)	CHARACTER	4	SZD_RIA_ TERMID	Terminal ID
(68)	CHARACTER	8	*	*reserved*
(70)	CHARACTER	80	SZD_ROA	Request output area
(70)	FULLWORD	4	SZD_ROA_FDBK1	Feedback 1 (extra)
(74)	FULLWORD	4	SZD_ROA_FDBK2	Feedback 2 (RESP2)
(78)	HALFWORD	2	SZD_ROA_OUT1	Output 1
(78)	HALFWORD	2	SZD_ROA_ SESSNSTATUS	Session status
(78)	HALFWORD	2	SZD_ROA_ ENDSTATUS	End status
(78)	HALFWORD	2	SZD_ROA_ STSNSTATUS	STSN status
(78)	HALFWORD	2	SZD_ROA_ SERVSTATUS	Service status
(78)	CHARACTER	2	*	Attributes

Offset Hex	Type	Len	Name (Dim)	Description
(78)	CHARACTER	1	SZD_ROA_ INPUTCONTROL	Input control
(7A)	HALFWORD	2	SZD_ROA_OUT2	Output 2
(7A)	HALFWORD	2	SZD_ROA_ RESPSTATUS	Response status
(7A)	HALFWORD	2	SZD_ROA_ ACQSTATUS	Acquire status
(7A)	CHARACTER	1	SZD_ROA_ RESPONSE	DFHSTSTM response *
(7B)	CHARACTER	1	SZD_ROA_ REASON	DFHSTSTM reason *
(7C)	HALFWORD	2	SZD_ROA_OUT3	Output 3
(7C)	HALFWORD	2	SZD_ROA_ ALARMSTATUS	Alarm status
(7C)	HALFWORD	2	SZD_ROA_ FMHSTATUS	FMH status
(7C)	HALFWORD	2	SZD_ROA_ INSTLSTATUS	Install status
(7E)	HALFWORD	2	*	Output 4
(80)	CHARACTER	8	SZD_ROA_OUT5	Output 5
(80)	HALFWORD	2	SZD_ROA_DEVICE	Device type
(80)	CHARACTER	8	SZD_ROA_ JOURNALNAME	Journal name
(80)	HALFWORD	2	SZD_ROA_ STATE	Conversation state
(88)	HALFWORD	2	SZD_ROA_OUT6	Output 6
(88)	HALFWORD	2	SZD_ROA_FORMAT	Data format
(88)	HALFWORD	2	SZD_ROA_ MSGJRNL	Journal control
(8C)	FULLWORD	4	SZD_ROA_RES1	Result 1
(8C)	FULLWORD	4	SZD_ROA_FIELDS	Field count
(8C)	FULLWORD	4	SZD_ROA_ ACQNUM	Acquire count
(8C)	FULLWORD	4	SZD_ROA_ SENSEDATA	Sense data
(8C)	FULLWORD	4	SZD_ROA_ ESMRESP	ESM response
(90)	FULLWORD	4	SZD_ROA_RES2	Result 2
(90)	FULLWORD	4	SZD_ROA_ DATALEN	Data length
(90)	FULLWORD	4	SZD_ROA_ CONVNUM	Conversation count
(90)	FULLWORD	4	SZD_ROA_ ESMREASON	ESM reason
(94)	FULLWORD	4	SZD_ROA_RES3	Result 3
(94)	FULLWORD	4	SZD_ROA_ REMFLENGTH	Remaining length
(94)	FULLWORD	4	SZD_ROA_ CURSOR	Cursor
(98)	FULLWORD	4	SZD_ROA_RES4	Result 4
(98)	FULLWORD	4	SZD_ROA_LINES	Line count
(98)	FULLWORD	4	SZD_ROA_ SEQNUMIN	Inbound seq num
(98)	FULLWORD	4	SZD_ROA_ WAITCONVNUM	Wait-conv count
(98)	FULLWORD	4	SZD_ROA_ POSITION	Position
(9C)	FULLWORD	4	SZD_ROA_RES5	Result 5
(9C)	FULLWORD	4	SZD_ROA_ COLUMNS	Column count
(9C)	FULLWORD	4	SZD_ROA_ SEQNUMOUT	Outbound seq num
(9C)	FULLWORD	4	SZD_ROA_ LASTACQCODE	Last acquire code
(9C)	FULLWORD	4	SZD_ROA_ SIZE	Size
(A0)	CHARACTER	8	SZD_ROA_OUC1	Outchar 1
(A0)	CHARACTER	8	SZD_ROA_CONVID	Conv ID
(A0)	CHARACTER	8	SZD_ROA_POOL	Pool
(A0)	CHARACTER	8	SZD_ROA_APPL	Appl name
(A0)	CHARACTER	8	SZD_ROA_ PASSTICKET	Passticket
(A8)	CHARACTER	8	SZD_ROA_OUC2	Outchar 2
(A8)	CHARACTER	8	SZD_ROA_TARGET	Target
(B0)	CHARACTER	8	SZD_ROA_OUC3	Outchar 3
(B0)	CHARACTER	8	SZD_ROA_NODE	Node
(B0)	CHARACTER	8	SZD_ROA_ PROPERTYSET	Property set
(B0)	CHARACTER	8	SZD_ROA_ ATTRS	Attributes
(B0)	CHARACTER	1	SZD_ROA_ COLOR	- colour
(B1)	CHARACTER	1	SZD_ROA_ HILIGHT	- highlighting

Offset Hex	Type	Len	Name (Dim)	Description
(B2)	CHARACTER	1	SZD_ROA_ VALIDATION	- validation
(B3)	CHARACTER	1	SZD_ROA_PS	- PS
(B4)	CHARACTER	1	SZD_ROA_ OUTLINE	- outlining
(B5)	CHARACTER	1	SZD_ROA_ TRANSPARENCY	- transparency
(B6)	CHARACTER	1	SZD_ROA_ BACKGROUND	- background
(B7)	CHARACTER	1	SZD_ROA_ FIELDATTR	- field
	11..		*	-
	..1.		SZD_ROA_ PROTECT	- protect
	...1 111.		*	-
	1		SZD_ROA_ MDT	- MDT
(B8)	CHARACTER	8	*	reserved
(C0)	CHARACTER		*	End of RPA

Constants

Len	Type	Value	Name	Description
4	DECIMAL	192	DFHSZDRP_LEN	
= FEPI Resource Manager Request Subtype Codes =				
2	DECIMAL	0	SZD_RIA_REQSUB_ NULL	nodsubtype
2	DECIMAL	4	SZD_RIA_REQSUB_FMT	formatted data
2	DECIMAL	8	SZD_RIA_REQSUB_ DATA	Datastream
2	DECIMAL	4	SZD_RIA_REQSUB_ CONV	Conversation
2	DECIMAL	8	SZD_RIA_REQSUB_ STSN	STSN
2	DECIMAL	12	SZD_RIA_REQSUB_FLD	FIeld
2	DECIMAL	4	SZD_RIA_REQSUB_TGT	Target
2	DECIMAL	8	SZD_RIA_REQSUB_ NODE	Node
2	DECIMAL	12	SZD_RIA_REQSUB_ POOL	Pool
2	DECIMAL	16	SZD_RIA_REQSUB_ PCHG	Add/Delete pool
2	DECIMAL	20	SZD_RIA_REQSUB_ PROP	Properties
2	DECIMAL	24	SZD_RIA_REQSUB_ CONN	Connection
2	DECIMAL	4	SZD_RIA_REQSUB_ CTRL	Control

FEP18 Session control request block

CONTROL BLOCK NAME = DFHSZDSC
DESCRIPTIVE NAME = CICS (FEPI) Session Control Request Block
FUNCTION =
  Defines the Session Control Request Block.
  This data area is a part of the FEPI Resource Manager.
  It defines the format of the Session Control Request Block.
Lifetime = While a VTAM request is active
Storage class = 31-bit addressable
Location = Chained from Common block
Inner control blocks = Not applicable
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS = None
 MODULE TYPE = Data Area
EXTERNAL REFERENCES = None
  DATA AREAS = None
  CONTROL BLOCKS = None
  GLOBAL VARIABLES = None

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	284	DFHSZDSC	
(0)	CHARACTER	32	SZD_SC_EYE	
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	BITSTRING	8	SZD_SC_QEB	
(20)	ADDRESS	4	*	unused - available
(24)	ADDRESS	4	SZD_SC_QNEXT	Points to next in chain
(28)	CHARACTER	8	*	
(28)	FULLWORD	4	SZD_SC_REQTYPE	reserved
(2C)	BITSTRING	4	SZD_SC_FLAGS	reserved
(30)	HALFWORD	2	SZD_SC_TRINTVL	timer retry interval
(32)	HALFWORD	2	SZD_SC_TRTYPE	timer retry type
(34)	ADDRESS	4	SZD_SC_DYNAA	unused available
(38)	ADDRESS	4	SZD_SC_CM	common area ptr
(3C)	ADDRESS	4	SZD_SC_CD	connection ptr
(40)	ADDRESS	4	SZD_SC_ND	node area ptr
(44)	FULLWORD	4	SZD_SC_DYNAL	unused available
(48)	CHARACTER	212	SZD_SC_RPL	VTAM RPL + buffer
(48)	AREA	212	SZD_SC_VTAM	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	284	DFHSZDSC_LEN	

FEP19 Terminal simulation facility

CONTROL BLOCK NAME = DFHSZDSR
DESCRIPTIVE NAME = CICS (FEPI) Terminal Simulation Facility
FUNCTION = Identifies the nodes and targets associated
with a given resource pool.
One DSR is created for each node and target
associated with each pool. It contains a pointer
to either a node or target (depending upon which
it represents)
LIFETIME = for the life of a node-pool or target-pool
association. Created during INSTALL POOL/ ADD POOL
processing, and deleted as a result of DISCARD POOL,
DISCARD NODE, DISCARD TARGET or DELETE POOL
processing.
STORAGE CLASS = 31-bit addressable
LOCATION = The DSR may be located from the DPD, DND or DTD
data areas.
INNER CONTROL BLOCKS =
DFHSZDEC eyecatcher data structure.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS =
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	84	DFHSZDSR	
(0)	CHARACTER	32	SZD_SR_EYE	eye catcher
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	

These fields chain the DSR off the pool with which the resource is being associated. There are 2 queues. One for nodes, and one for targets.

Offset Hex	Type	Len	Name (Dim)	Description
(20)	ADDRESS	4	SZD_SR_PREV	previous in pool
(24)	ADDRESS	4	SZD_SR_NEXT	next in pool

These fields chain the DSR off the resource to which it relates. This may be either a node or a target.

Offset Hex	Type	Len	Name (Dim)	Description
(28)	ADDRESS	4	SZD_SR_ORPREV	prev on resource
(2C)	ADDRESS	4	SZD_SR_ORNEXT	next on resource

This is the pool that owns the DSR

Offset Hex	Type	Len	Name (Dim)	Description
(30)	ADDRESS	4	SZD_SR_PDPTR	owning pool

This is the address of the resource being represented.

Offset Hex	Type	Len	Name (Dim)	Description
(34)	ADDRESS	4	SZD_SR_TDPTR	owning target,
(34)	ADDRESS	4	SZD_SR_NDPTR	or owning node
(38)	FULLWORD	4	SZD_SR_USAGE	resource usage counter

Statistics counters - used by target surrogate only

Offset Hex	Type	Len	Name (Dim)	Description
(3C)	FULLWORD	4	SZD_SR_NODES	Used during stats collection
(40)	FULLWORD	4	SZD_SR_ TOTALLOCATES	Total # conversation allocates
(44)	FULLWORD	4	SZD_SR_ALLOCATESWAITING	Current # allocates waiting
(48)	FULLWORD	4	SZD_SR_PKALLOCATESWAITING	Peak # allocates waiting
(4C)	FULLWORD	4	SZD_SR_TOTALLOCATEWAITS	Total # allocates waited
(50)	FULLWORD	4	SZD_SR_TIMEOUTS	# allocates that timed out

Constants

Len	Type	Value	Name	Description
4	DECIMAL	84	DFHSZDSR_LEN	

FEP20 Target descriptor

```
  CONTROL BLOCK NAME = DFHSZDTD
  DESCRIPTIVE NAME = CICS (FEPI) Target descriptor
  FUNCTION = Contains the information needed by the resource
             manager to represent and control activity with
             a back-end application. One control block exists
             for each target defined by the installation during
             INSTALL processing.
  LIFETIME = Created by DFHSZRIT during INSTALL processing.
             Deleted by DFHSZRDT during DISCARD processing.
  STORAGE CLASS = 31-bit addressable.
  LOCATION = Located from the DFHSZDCM.
  INNER CONTROL BLOCKS =
  NOTES :
   DEPENDENCIES = S/370
   RESTRICTIONS =
   MODULE TYPE = Control block definition
  EXTERNAL REFERENCES =
    DATA AREAS =
    CONTROL BLOCKS = DFHSZDEC (Eyecatcher structure definition)
    GLOBAL VARIABLES (Macro pass) =
 & NOTE
 & The first portion of DFHSZDTD is structured to be identical to
 & the first portion of the DQE. This MUST not change. If changes
 & are made to the DQE, then this area must be updated to match.
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	184	DFHSZDTD	
(0)	CHARACTER	32	SZD_TD_EYE	Request parm area
(0)	HALFWORD	2	SZD_EC_LENGTH	
(2)	CHARACTER	1	SZD_EC_GT	
(3)	CHARACTER	8	SZD_EC_NAME	
(B)	CHARACTER	5	*	
(10)	CHARACTER	8	SZD_EC_SPID	
(18)	UNSIGNED	4	SZD_EC_CBID	
(1C)	CHARACTER	4	*	
(20)	CHARACTER	24	SZD_TD_WE	
(20)	BITSTRING	8	SZD_TD_QCB	Target DQE
(20)	ADDRESS	4	SZD_TD_QP	Previous entry
(24)	ADDRESS	4	SZD_TD_QC	Next queue element
(28)	FULLWORD	4	SZD_TD_REQ	Request type
(2C)	BITSTRING	4	*	request flags
	1...		*	reserved - not avail
	.1..		*	reserved - not avail
	..1.		SZD_TD_ON_Q	ON THE Process Q
	...1		SZD_TD_ON_QIRB	ON THE IRB Process Q
	.... 1...		SZD_TD_ON_TMR	on the timer queue
	.... .1..		*	reserevd - not avail
	.... ..1.		*	reserved - not avail
	1		*	reserved - not avail

NOTE
End of section that must match DFHSZDQE

Offset Hex	Type	Len	Name (Dim)	Description
(30)	HALFWORD	2	SZD_TD_TRINTVL	Timer retry interval
(32)	HALFWORD	2	SZD_TD_TRTYPE	Retry type required
(34)	FULLWORD	4	*	unused available

Target control flags

Offset Hex	Type	Len	Name (Dim)	Description
(38)	BITSTRING	4	SZD_TD_CS_FLAGS	
	1...		*	unused - available
	.1..		*	unused - available
	..1.		SZD_TD_REQ_FAIL	REQSESS failed
(3C)	CHARACTER	4	SZD_TD_DEFTRAN	saved tranid @BA65235C

When REQSESS processing is required for a connection, it is queued here, and the target is queued to the resource manager for processing (unless already queued).

Offset Hex	Type	Len	Name (Dim)	Description
(40)	BITSTRING	8	SZD_TD_RE_QCB	REQSESS Q
(40)	ADDRESS	4	SZD_TD_RE_QC	FIRST ENTRY
(44)	ADDRESS	4	SZD_TD_RE_CTR	POOL CTR

Offset Hex	Type	Len	Name (Dim)	Description
This is the configuration management portion of the target.				
(48)	CHARACTER	112	SZD_TD_API	
(48)	ADDRESS	4	SZD_TD_PREV	Prev. target
(4C)	ADDRESS	4	SZD_TD_NEXT	Next target
(50)	ADDRESS	4	SZD_TD_SRLIST	Surrogate list
(54)	ADDRESS	4	SZD_TD_CDLIST	Connection list
(58)	CHARACTER	8	SZD_TD_NAME	FEPI resource name
(60)	CHARACTER	8	SZD_TD_PLUN	network AM rsrc name
(68)	HALFWORD	2	SZD_TD_ SERVSTATUS	service status
(6A)	HALFWORD	2	SZD_TD_ INSTSTATUS	Installation status
(6C)	FULLWORD	4	SZD_TD_CURRENT	Usage counter
(70)	FULLWORD	4	SZD_TD_USAGE	Usage counter
(74)	FULLWORD	4	SZD_TD_RCOUNT	Usage counter
(78)	CHARACTER	64	SZD_TD_UDATA	User data

Constants

Len	Type	Value	Name	Description
4	DECIMAL	184	DFHSZDTD_LEN	

FEP21 Frontend programming interface

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	320	DFHSZSPS	
(0)	HALFWORD	2	SZSEYEL	CB Length
(2)	CHARACTER	14	SZSEYEC	Eyecatcher
== ==				
(10)	UNSIGNED	4	SZS_SYSSTATE	FEPI Status
== = TCB Operation Controls = ==				
(14)	UNSIGNED	2	SZSTMODE	TCB for RM running
(16)	UNSIGNED	2	SZSTLEV	TCB RM Trigger
== = Unused Storage = ==				
(18)	UNSIGNED	4	*	Unused
(1C)	CHARACTER	3	*	Unused
=== = Flag byte ===				
(1F)	BITSTRING	1	*	Misc flags
	1...		SZS_CONFDATA	CONFDATA on
	.111 1111		*	
== = FEPI Anchor points = ==				
(20)	ADDRESS	4	SZSANCCI	CICS Storage Anchor
(24)	ADDRESS	4	SZSANCRM	RM Storage Anchor
(28)	ADDRESS	4	*	
(2C)	ADDRESS	4	*	
== = FEPI Unused Storage = ==				
(30)	ADDRESS	4	*	
(34)	ADDRESS	4	*	
(38)	ADDRESS	4	*	
(3C)	ADDRESS	4	*	
== = FEPI Storage Sub-pool Tokens = ==				
(40)	CHARACTER	8	SZS_SP_AC	SPT for ACBs
(48)	CHARACTER	8	SZS_SP_CD	SPT for Conn Cont
(50)	CHARACTER	8	SZS_SP_CM	SPT for Common Cont
(58)	CHARACTER	8	SZS_SP_CV	SPT for Conv Cont
(60)	CHARACTER	8	SZS_SP_DA	SPT for Data Areas
(68)	CHARACTER	8	SZS_SP_DS	SPT for Device Supp

Offset Hex	Type	Len	Name (Dim)	Description
(70)	CHARACTER	8	SZS_SP_DT	SPT for Device Type
(78)	CHARACTER	8	SZS_SP_NB	SPT for NIBs
(80)	CHARACTER	8	SZS_SP_ND	SPT for Node Defs
(88)	CHARACTER	8	SZS_SP_PD	SPT for Pool Descs
(90)	CHARACTER	8	SZS_SP_PS	SPT for Prop Descs
(98)	CHARACTER	8	SZS_SP_RP	SPT for RPLs
(A0)	CHARACTER	8	SZS_SP_RQ	SPT for Requests
(A8)	CHARACTER	8	SZS_SP_TD	SPT for Target Descs
(B0)	CHARACTER	8	SZS_SP_WE	SPT for Work Eles
(B8)	CHARACTER	8	SZS_SP_SR	SPT for Surrogates
(C0)	CHARACTER	8	*	Unused
(C8)	CHARACTER	8	*	Unused
(D0)	CHARACTER	8	*	Unused
(D8)	CHARACTER	8	*	Unused
(E0)	CHARACTER	8	*	Unused
(E8)	CHARACTER	8	*	Unused
(F0)	CHARACTER	8	*	Unused
(F8)	CHARACTER	8	*	Unused
(100)	CHARACTER	8	*	Unused
(108)	CHARACTER	8	*	Unused
(110)	CHARACTER	8	*	Unused
(118)	CHARACTER	8	*	Unused
(120)	CHARACTER	8	*	Unused
(128)	CHARACTER	8	*	Unused
(130)	CHARACTER	8	*	Unused
(138)	CHARACTER	8	*	Unused
== = FEPI Control Block length = ==				
(140)	CHARACTER		SZSEND	End of Control Block

Constants

Len	Type	Value	Name	Description
4	DECIMAL	320	SZSLEN	Control Block Length
4	DECIMAL	0	SZS_SYSSTATE_NEVAC	Not yet accessed
4	DECIMAL	1	SZS_SYSSTATE_ CLOSED	Inactive
4	DECIMAL	2	SZS_SYSSTATE_ INITING	Starting
4	DECIMAL	3	SZS_SYSSTATE_OPEN	Running
4	DECIMAL	4	SZS_SYSSTATE_ TERM_NORM	Normal Shutdown
4	DECIMAL	5	SZS_SYSSTATE_ TERM_IMMED	Immediate Shutdown
4	DECIMAL	6	SZS_SYSSTATE_ TERM_FORCE	Forced Termination
4	DECIMAL	7	SZS_SYSSTATE_ FAILED	FEPI Abended
2	DECIMAL	1	SZSTMODE_QR	RM is always to run under the QR TCB
2	DECIMAL	2	SZSTMODE_SZ	RM is always to run under the SZ TCB
2	DECIMAL	3	SZSTMODE_DYNAMIC	RM will run under the QR | SZ TCB, depending on workload

FLLBC File control locks locator block

```
CONTROL BLOCK NAME = DFHFLLBC
DESCRIPTIVE NAME = CICS FC Locks Locator Block (FLLB)
FUNCTION =
     DFHFLLB describes the DSECT for the File Control Locks
     Locator Block. This block records a UOW that held locks
     for a Lost Locks data set or a UOW for which the 'override'
     condition exists for a data set it is using, or a UOW which
     made updates to an RLS file prior to an OFFSITE=YES restart
     being performed. The override, or 'NonRLSupdatePermitted',
     condition is returned by VSAM when a file is opened with
     RLS access for a dataset which has had its retained locks
     overriden by a non-RLS batch program. Offsite recovery
     occurs when a remote site recovery is performed which
     involves data sets that were open in RLS mode.
     In the case of the Lost Locks condition and for offsite
     recovery, FLLBS are created by DFHFCRR. In the case of
     the override condition, FLLBs are created by DFHFCO1
     immediately after a file open which has returned the
     'override' reason code.
     In all cases the FLLBs are chained from both the
     associated DSNB and the associated FRAB.
     The address of the head of the FLLB chain in the DSNB is at
     field FCTBC_FLLB_CHAIN, and in the FRAB is at field FRAB
     FRAB_FLLB_CHAIN_ADDRESS. There is one FLLB per file per UOW.
     FLLBs are getmained from the FLLB subpool which is
     created by DFHFCRP during File Control Initialisation.
     File Control Locks Locator Blocks are freemained by
     DFHFCRC when lost locks recovery has been completed or
     by DFHFCRC at commit time when there are no longer any
     flabs with retain_reason of not_retained for the dataset.
LIFETIME =
     In the case of Lost Locks condition :
      Created when processing lost locks at RLS restart.
      Deleted at end of Lost Locks Recovery.
     In the case of the override condition :
      Created when a file is opened for a data set
      that VSAM has recorded as having had retained locks
      overriden by a non-RLS batch program.
      Deleted at commit tume by DFHFCRC.
     In the case of offsite recovery :
      Created when processing an OFFSITE=YES RLS restart.
      Deleted at commit tume by DFHFCRC.
STORAGE CLASS =
     Above 16M line. CICS key.
LOCATION =
INNER CONTROL BLOCKS = None.
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS = None
 MODULE TYPE = Control block definition
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	41	DFHFLLB	
Eye catcher				
(0)	CHARACTER	16	FLLB_EYE_CATCHER	Eye catcher
(0)	UNSIGNED	2	FLLB_LENGTH	Length of FLLB
(2)	CHARACTER	6	FLLB_EYE1	>DFHFC FC 'domain'
(8)	CHARACTER	8	FLLB_EYE2	FLLB
Main part of FLLB				
(10)	CHARACTER	25	FLLB_MAIN_PART	Main part of FLLB
(10)	ADDRESS	4	FLLB_DSNB_ ADDRESS	DSNB address
(14)	ADDRESS	4	FLLB_NEXT_ IN_DSNB_CHAIN	Ptr to next FLLB in DSNB chain
(18)	ADDRESS	4	FLLB_PREV_ IN_DSNB_CHAIN	Pointer to previous FLLB in DSNB chain
(1C)	ADDRESS	4	FLLB_NEXT_ IN_FRAB_CHAIN	Pointer to next FLLB in FRAB chain
(20)	CHARACTER	8	FLLB_LUWID	LUWID
(28)	BITSTRING	1	FLLB_LOCK_ CONDITION	Lock Condition
	1...		FLLB_LOST_ LOCKS	Lost Locks
	.1..		FLLB_OVERRIDEN_ LOCKS	Overriden Locks

Offset Hex	Type	Len	Name (Dim)	Description
	..1.		FLLB_OFFSITE_RECOVERY	Offsite recovery
	...1 1111		*	Reserved

KCB Kernel anchor block

```
CONTROL BLOCK NAME = DFHKEGBL
DESCRIPTIVE NAME = CICS (KE) Kernel Global.
FUNCTION =
     Kernel's Anchor for all other control blocks.
     This anchor points to kernel programs, domain and task
     tables.
     These blocks are described in DFHKECB.
     The Kernel Anchor is addressed in two ways:
     First, if the Kernel is Called the R13 -> Linkage that
     identifies the Kernel Global.
     Secondly, the KCB can be addressed from the AFCS via low
     core, the TCB Extension and the AFCB.
     The AFCS/AFCB/AFT is defined in DFHAFCP, a PLAS copy book.
LIFETIME = One per Space, for the duration of the CICS Run.
STORAGE CLASS =
LOCATION = See Above.
INNER CONTROL BLOCKS =
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS =
 MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
  DATA AREAS =
  CONTROL BLOCKS =
  GLOBAL VARIABLES (Macro pass) =
  Kernel Global Storage
    Global to this CICS Step
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	528	DFHKCB	
(0)	CHARACTER	68	KCB_PROCESS_OWN	Process own table
(0)	CHARACTER	16	KCB_PREFIX	Standard prefix
(0)	HALFWORD	2	KCB_LENGTH	Length of KCB
(2)	CHARACTER	1	KCB_ARROW	>
(3)	CHARACTER	3	KCB_DFH	DFH
(6)	CHARACTER	2	KCB_DOMID	KE
(8)	CHARACTER	8	KCB_BLOCK_NAME	KCB
(10)	ADDRESS	4	KCB_DOMAIN_CALL	Domain call
(14)	ADDRESS	4	KCB_PERCOLATE	Percolate
(18)	ADDRESS	4	KCB_DOMAIN_RETURN	Domain return
(1C)	ADDRESS	4	KCB_RECOVERY_EXIT	Recovery Exit
(20)	ADDRESS	4	KCB_RECOVERY_REQUEST	Recovery Request
(24)	ADDRESS	4	KCB_RESET_ADDRESS	Reset Address
(28)	ADDRESS	4	KCB_SUBROUTINE_CALL	Subroutine call
(2C)	ADDRESS	4	KCB_SUBROUTINE_RETURN	Subroutine return
(30)	ADDRESS	4	KCB_TRACE_DOM_CALL	Address of DFHTRPX, Fast Trace Module
(34)	ADDRESS	4	KCB_TRACE_DOM_TABLE	Address of Trace Global Storage
(38)	ADDRESS	4	KCB_DOMAIN_RETURN_24	Dom. ret. from smode
(3C)	ADDRESS	4	KCB_SUBROUTINE_RETURN_24	Sub. ret. from smode 24@L2A
(40)	ADDRESS	4	KCB_ADD_CICS_RECOVERY_EP	DFHKESTX entry point
(44)	FULLWORD	4	KCB_TEMP_STATIC_TASK_NUMBER	
	Number of temporary static tasks			

Offset Hex	Type	Len	Name (Dim)	Description
(48)	UNSIGNED	4	KCB_RUNAWAY_ LIMIT	System runaway limit
(4C)	ADDRESS	4	KCB_OVERFLOW_STACK_LM_LOCK	Lock for queuing tasks if low on 31-overflow stacks
(50)	UNSIGNED	2	*	Reserved
(52)	UNSIGNED	2	KCB_MIN_FREE_OVERFLOWS	Minimum no. of 31-overflow stacks to maintain
Kernel status fields				
(54)	BITSTRING	4	KCB_KERNEL_ STATUS	Kernel status fields
(54)	BITSTRING	1	KCB_JOB_ STEP_STATUS	Status of CICS Job Step
	1...		KCB_TERMINATE_REQUESTED	Terminate CICS requested
	.1..		KCB_DUMP_REQUESTED	MVS Sdump requested
	..1.		KCB_CANCEL_REQUESTED	X22 Abend has occurred
	...1		KCB_NORMAL_TERMINATION	Normal term. requested
	.... 1...		KCB_OUT_ OF_STACK	Out of stack space
	.... .111		*	Reserved
(55)	BITSTRING	1	KCB_FACILITY_ STATUS	Status of Kernel facilities
	1...		KCB_QUIESCE_DOMAIN_RECEIVED	KE has been told to quiesce
	.1..		KCB_ESTAE_ ACTIVE	Estae active
	..1.		KCB_HPO_ACTIVE	HPO available
	...1 11..		*	RESERVED
	.... ..1.		KCB_TRAP_ ACTIVE	Kernel global trap active
	1		KCB_CICS	0-current job is STUP 1-current job is CICS
(56)	BITSTRING	1	KCB_TIMER_ STATUS	Kernel timer status
	1...		*	Reserved
	.1..		KCB_CLOCKING_ ACTIVE	CPU time recording active
	..1.		KCB_STIMER_ ACTIVE	Kernel STimer active
	...1 1111		*	Reserved
(57)	BITSTRING	1	*	Reserved
Kernel table addresses.				
(58)	ADDRESS	4	KCB_TASK_ CHAIN_START	Address of first task in global chain
(5C)	ADDRESS	4	*	Reserved
(60)	CHARACTER	8	KCB_SEG24_ QUICK_CELL	24-bit segment q-c chain
(60)	ADDRESS	4	KCB_SEG24_ FIRST_FREE	First free 24-bit segment
(64)	FULLWORD	4	KCB_SEG24_GUARD	Quick-cell guard count
(64)	UNSIGNED	2	KCB_SEG24_GUARD_COUNT	Half-word guard count for free segment chain
(66)	UNSIGNED	2	KCB_SEG24_FREE_SEGS	Number of free segments in chain
(68)	CHARACTER	8	KCB_SEG31_ QUICK_CELL	31-bit segment q-c chain
(68)	ADDRESS	4	KCB_SEG31_ FIRST_FREE	First free 31-bit segment
(6C)	FULLWORD	4	KCB_SEG31_GUARD	Quick-cell guard count
(6C)	UNSIGNED	2	KCB_SEG31_GUARD_COUNT	Half-word guard count for free segment chain
(6E)	UNSIGNED	2	KCB_SEG31_FREE_SEGS	Number of free segments in chain
(70)	ADDRESS	4	KCB_DOMAIN_TABLE	Address of domain table header
(74)	ADDRESS	4	*	Reserved
(78)	ADDRESS	4	KCB_ERROR_TABLE	Address of error table header
(7C)	ADDRESS	4	KCB_KTCB_TABLE	Address of KTCB table header
Kernel global data.				
(80)	CHARACTER	8	KCB_STIMER_ INTERVAL	MVS STIMER interval
(88)	FULLWORD	4	KCB_DOMAIN_ NUMBER	Number of domains
(8C)	FULLWORD	4	KCB_GATE_NUMBER	Number of gates
(90)	FULLWORD	4	KCB_STATIC_TASK_NUMBER	Number of static tasks
(94)	HALFWORD	2	KCB_DUMP_RETRY	SDUMP retry time

Offset Hex	Type	Len	Name (Dim)	Description
(96)	BITSTRING	1	KCB_GLOBAL_ DATA_FLAGS	Various flags
	1...		KCB_ISC_ AVAILABLE	ISC is available in this system
	.1..		KCB_XRF	XRF option
	..1.		KCB_STORAGE_ PROTECT_SUPPORTED	Hardware supports storage protect
	...1		KCB_SET_ DUB_ISSUED	SetDubDefault issued
	 1111		*	Reserved
(97)	CHARACTER	1	*	Reserved
(98)	CHARACTER	8	*	Reserved
(98)	FULLWORD	4	*	Reserved
(9C)	BITSTRING	4	*	Reserved
(A0)	FULLWORD	4	KCB_KTCB_NUMBER	Number of KTCBs
(A4)	CHARACTER	4	KCB_TIMER_STATE	Status of CPU timing, communicates between the different KTCBs
	1...		KCB_TIMER_ ACTIVE	CPU timing is active
(A4)	BITSTRING	1	*	Padding
(A6)	HALFWORD	2	KCB_TIMER_ CHANGES	Number of times state has changed
(A8)	CHARACTER	8	KCB_PARMS	OS parameters
(A8)	ADDRESS	4	KCB_PARMS_ADDR	Address of data
(AC)	FULLWORD	4	KCB_PARMS_LEN	Length of data
(B0)	ADDRESS	4	*	Unused
(B4)	CHARACTER	48	KCB_DESCRIPTION	Address space descriptions
(B4)	CHARACTER	8	KCB_GENERIC_ APPLID	VTAM applid
(BC)	CHARACTER	8	KCB_SPECIFIC_ APPLID	VTAM applid
(C4)	CHARACTER	8	KCB_XRF_ COMMAND_LIST	Name of failure commands
(CC)	CHARACTER	8	KCB_ALTERNATE_ XRF_IDS	AXI table name
(D4)	CHARACTER	4	KCB_SYSID	System entry name
(D8)	CHARACTER	8	KCB_SIT_NAME	System Initialisation table
(E0)	CHARACTER	1	KCB_OP_SYS	Operating system (X=MVS/XA)
(E1)	CHARACTER	1	KCB_OP_VERSION	Version of above system
(E2)	CHARACTER	1	KCB_OP_RELEASE	Release of above system
(E3)	CHARACTER	1	KCB_OP_ MODIFICATION	Modification of above systm
(E4)	ADDRESS	4	KCB_IPL_STACK	First system stack
(E8)	ADDRESS	4	KCB_MODULE_ VECTOR_POINTER	Critical Csect pointer
(EC)	ADDRESS	4	KCB_WINDOW_ VECTOR_POINTER	Windows pointer
(F0)	HALFWORD	2	*	Reserved
(F2)	UNSIGNED	1	KCB_CICS_SVC	The CICS Service SVC
(F3)	UNSIGNED	1	KCB_CICS_ SVC_NUMBER	CICS Service SVC number
(F4)	CHARACTER	8	KCB_LOCAL_ TIME_DELTA	Diffrnce between STCK & TOD
(F4)	UNSIGNED	4	KCB_DELTA_HIGH	High order word
(F8)	UNSIGNED	4	KCB_DELTA_LOW	Low order word
(FC)	BITSTRING	1	KCB_GMT_TO_LOCAL	Indicates how to re-instate local time from GMT
	1...		KCB_ADD_DELTA	Add delta to STCK time
	.1..		KCB_SUBTRACT_ DELTA	Subtract delta from STCK
	..11 1111		*	Unused
(FD)	BITSTRING	1	KCB_DATE_FORMAT	CICS default date format
	1...		KCB_YYMMDD	Date format YYMMDD
	.1..		KCB_DDMMYY	Date format DDMMYY
	..1.		KCB_MMDDYY	Date format MMDDYY
	...1 1111		*	Padding
(FE)	BITSTRING	1	KCB_NOTIFY_ RESET_DOMAINS	
	1...		KCB_NOTIFY_ TRACE	Trace Domain to be notified
	.111 1111		*	Unused
(FF)	UNSIGNED	1	*	Padding
(100)	FULLWORD	4	KCB_TRACE	Trace management data
(100)	BITSTRING	1	KCB_TRMF	Trace master flags
	1...		KCB_MASTER	...Master flag
	.1..		KCB_SYSTEM_ MASTER	...System master flag
(101)	UNSIGNED	1	*	Padding
(102)	HALFWORD	2	KCB_TRACE_COUNT	Trace data change count
(104)	CHARACTER	12	KCB_TRAP	Global trap field
(104)	BITSTRING	1	KCB_TRAP_STATUS	Status of global trap
	1...		KCB_TRAP_ ENABLED	SET_TRAP has been issued, so address+parameter valid
	.111 1111		*	Padding
(105)	CHARACTER	3	*	Padding
(108)	ADDRESS	4	KCB_TRAP_ ADDRESS	Address to call
(10C)	ADDRESS	4	KCB_TRAP_ PARAMETER	

Offset Hex	Type	Len	Name (Dim)	Description
				Address to pass
(110)	ADDRESS	4	KCB_DFHCRC_ ADDRESS	Need this for Estaes
(114)	FULLWORD	4	KCB_MXT_ EXTRA_SEGMENTS_24	Extra non-disposable 24-bit segments to support current MXT value
(118)	CHARACTER	8	KCB_STATIC_ QUICK_CELL	Static quick-cell chn
(118)	ADDRESS	4	KCB_STATIC_ FIRST_FREE	First task in free list@L4A
(11C)	FULLWORD	4	KCB_STATIC_ GUARD	Quick-cell guard count
(120)	CHARACTER	8	KCB_DYNAMIC_ QUICK_CELL	Dynamic q-c chain
(120)	ADDRESS	4	KCB_DYNAMIC_ FIRST_FREE	First task in free list@L4A
(124)	FULLWORD	4	KCB_DYNAMIC_ GUARD	Quick-cell guard count
(128)	ADDRESS	4	KCB_DISPOSAL_ CHAIN	Start of disposal chain
(12C)	FULLWORD	4	KCB_EXCESS_ STATIC_TASKS	Static tasks surplus to requirements but not yet on the disposal chain
(130)	CHARACTER	8	KCB_STK24_ SUBPOOL_TOKEN	Subpool for initial 24-bit stack segments
(138)	CHARACTER	8	KCB_STK31_ SUBPOOL_TOKEN	Subpool for initial 31-bit stack segments
(140)	CHARACTER	8	KCB_STK24E_ SUBPOOL_TOKEN	Subpool for extra 24-bit stack segments
(148)	CHARACTER	8	KCB_STK31E_ SUBPOOL_TOKEN	Subpool for extra 31-bit stack segments
(150)	CHARACTER	8	KCB_TASK_ SUBPOOL_TOKEN	Subpool for Kernel tasks
(158)	CHARACTER	8	KCB_KE_LOCK	Kernel global lock
(160)	FULLWORD	4	KCB_MXT_ EXTRA_SEGMENTS_31	Extra non-disposable 31-bit segments to support current MXT value
(164)	CHARACTER	20	*	Reserved
(178)	ADDRESS	4	KCB_DOMAIN_ VECTOR (0 36)	Optimised route to domain table entries
(210)	CHARACTER		KCB_DOMAIN_ TABLE_START	Round to dword

Module Vector Pointer.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	KCB_MODULE_VECTOR	Pointers to critical addresses
(0)	FULLWORD	4	KCB_VECTOR_SIZE	Number of entries
(4)	FULLWORD	4	*	Padding
(8)	CHARACTER	8	KCB_VECTOR_ENTRY (6)	Critical vector entries *
(8)	ADDRESS	4	KCB_MODULE_ ADDRESS	Address of Module
(C)	FULLWORD	4	KCB_MODULE_ LENGTH	Length of Module
(38)	CHARACTER		*	Round to double-word

KECB Kernel control blocks

CONTROL BLOCK NAME = DFHKECB
DESCRIPTIVE NAME = CICS (KE) Kernel Control Blocks.
FUNCTION =
LIFETIME = All storage described here is long-life.
STORAGE CLASS = MVS Getmained.
LOCATION = Above the line, except for 24-bit stack entries.
INNER CONTROL BLOCKS =
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS =
 MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
  DATA AREAS =
  CONTROL BLOCKS =
  GLOBAL VARIABLES (Macro pass) =
  Domain Table Header

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(0)	STRUCTURE	32	DOMAIN_HEADER	Domain table header
(0)	CHARACTER	16	DOH_PREFIX	Standard prefix
(0)	HALFWORD	2	DOH_LENGTH	Length of table header
(2)	CHARACTER	1	DOH_ARROW	>
(3)	CHARACTER	3	DOH_DFH	DFH
(6)	CHARACTER	2	DOH_DOMID	KE
(8)	CHARACTER	8	DOH_BLOCK_NAME	DOH
(10)	ADDRESS	4	DOH_TABLE_START	First domain table entry
(14)	ADDRESS	4	DOH_TABLE_END	End of domain table
(18)	HALFWORD	2	DOH_ENTRY_LENGTH	Length domain table entry
(1A)	HALFWORD	2	*	Reserved
(1C)	ADDRESS	4	*	Reserved
(20)	CHARACTER		DOH_END	Round to double-word

Domain Table Entry

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(0)	STRUCTURE	256	DOMAIN_ENTRY (0 36)	
(0)	CHARACTER	8	DOM_NAME	Domain name
(8)	FULLWORD	4	DOM_INDEX	Domain index
(C)	CHARACTER	4	DOM_STATE	Domain state flags
(C)	BITSTRING	1	DOM_STATE_FLAG	Domain state
	1...		DOM_TERMINATED	Domain terminated
	.111 1111		*	Reserved
(D)	BITSTRING	1	DOM_AFFINITY	
	1...		DOM_AFFINITY_ STEP	Affinity with Step TCB
	.1..		DOM_AFFINITY_ RO	Affinity with RO TCB
	..1.		DOM_AFFINITY_ QR	Affinity with QR TCB
	...1		DOM_AFFINITY_ CO	Affinity with CO TCB
	.... 1...		DOM_AFFINITY_ FO	Affinity with FO TCB
	.... .111		*	Reserved
(E)	BITSTRING	1	*	Reserved
(F)	BITSTRING	1	*	Reserved
(10)	ADDRESS	4	DOM_ANCHOR	Domain's global storage
(14)	BITSTRING	4	DOM_STANDARD_ TRACE	Std trace bits
(18)	BITSTRING	4	DOM_SPECIAL_ TRACE	Special trace bits
(1C)	FULLWORD	4	DOM_DEFAULT_ RECOVERY	Default recovery routine
(20)	CHARACTER	8	DOM_GATE_ TABLE_NAME	Gate table eye-catcher
(28)	CHARACTER	4	DOM_GATE_TABLE (0 53)	
(28)	ADDRESS	4	DOM_GATE_ENTRY	Gate entry point
(100)	CHARACTER		*	

Task

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	896	TASK_ENTRY	Task
(0)	CHARACTER	8	TAS_NAME	Eye-catcher TASENTRY
(8)	ADDRESS	4	TAS_NEXT_FREE	Free list pointer
(C)	FULLWORD	4	TAS_INDEX	Index of task entry
(10)	CHARACTER	12	TAS_STACK_ POINTERS	Pointers to task's stacks
(10)	ADDRESS	4	TAS_SEGMENT_ENTRY_31	Address of first segment for above-the-line segments
(14)	ADDRESS	4	TAS_SEGMENT_ENTRY_24	Address of first segment for below-the-line segments
(18)	ADDRESS	4	TAS_CURRENT_ STACK	Current stack of this task
(1C)	ADDRESS	4	TAS_FREE_SEGS_24	Free segment chain
(20)	ADDRESS	4	TAS_MONITORING_ TOKEN	Field used by monitoring
(24)	FULLWORD	4	TAS_ATTACH_TOKEN	Attach request token
(28)	ADDRESS	4	TAS_TCA_ADDRESS	TCA address
(2C)	CHARACTER	16	TAS_SEGMENT_ POINTERS	Pointers to task's segments
(2C)	ADDRESS	4	TAS_END_OF_SEGMENT_31	Last byte + 1 of segment
(30)	ADDRESS	4	TAS_CURRENT_STACK_31	Top 31-bit stack
(34)	ADDRESS	4	TAS_END_OF_SEGMENT_24	Last byte + 1 of segment
(38)	ADDRESS	4	TAS_CURRENT_STACK_24	Top 24-bit stack
(3C)	UNSIGNED	4	TAS_STATE	State of task
	1...		TAS_STATE_ ALLOCATED	Task is in use
	.1..		TAS_STATE_ DYNAMIC	Dynamic=1, Static=0
	..1.		TAS_STATE_ SPECIAL	Special tracing required
	...1		TAS_STATE_ STANDARD	Standard tracing required
	.... 1...		TAS_STATE_SUPPRESSED	Only exception tracing
	.... .1..		TAS_STATE_ DISPOSABLE	Disposable
	.... ..1.		TAS_STATE_ACQUIRED_FROM_SM	Acquired from SM
	1		TAS_STATE_LINKAGE_ERROR	Task has suffered an AKEG abend
(3D)	1...		TAS_STATE_TEMP_STATIC	Temporary static
(40)	ADDRESS	4	TAS_KTCB_ENTRY	Current KTCB entry for task
(44)	HALFWORD	2	TAS_TRACE_COUNT	Level of trace data in stack
(46)	HALFWORD	2	TAS_ERROR_COUNT	Number of stack entries marked as "in error"
(48)	FULLWORD	4	TAS_DOMAIN_INDEX	Domain index over TCB Attach
(4C)	CHARACTER	64	TAS_REGISTER_ STORAGE	Register save area -storage
(4C)	ADDRESS	4	TAS_REGISTER_ SAVE (16)	Register save area - array
(8C)	ADDRESS	4	TAS_FREE_SEGS_31	31 bit free seg chain
(90)	CHARACTER	16	TAS_CPU_CLOCK	Task clocking
(90)	CHARACTER	8	TAS_TOTAL_TIME	CPU time used so far
(98)	HALFWORD	2	TAS_RUNAWAY_ LEFT	# of intervals left
(9A)	BITSTRING	1	TAS_CLOCK_ STATUS	Clock status fields
	1...		TAS_CLOCK_ ACTIVE	CPU recording is active
	.1..		TAS_RUNAWAY_ ACTIVE	Runaway detection active
	..1.		TAS_RUNAWAY_EXPIRED	Runaway has occurred
	...1		TAS_RUNAWAY_STATE_INITIALISED	Runaway detection has been initialised for this execution slice
	.... 1...		TAS_SYSTEM_RUNAWAY	This task is using system runaway limit
	.... .1..		TAS_RUNAWAY_STOPPED	Runaway detection has been stopped for this task
	11		*	Reserved
(9B)	BITSTRING	1	*	Reserved
(9C)	HALFWORD	2	TAS_STOP_ RUNAWAY	# of Stop Runaway Timer requests.

Offset Hex	Type	Len	Name (Dim)	Description
(9E)	HALFWORD	2	TAS_PURGE_PROTECTION_COUNT	# of Start Purge Protection requests, 0 = not protected
(A0)	ADDRESS	4	TAS_XM_TRANSACTION_TOKEN	XM transaction token
(A4)	ADDRESS	4	TAS_PREV_TASK	Global chain prev. task
(A8)	ADDRESS	4	TAS_NEXT_TASK	Global chain next task
(AC)	ADDRESS	4	TAS_INIT_SEG_24	Initial 24-bit segment
(B0)	ADDRESS	4	TAS_INIT_SEG_31	Initial 31-bit segment

reflected there also.

Offset Hex	Type	Len	Name (Dim)	Description
(B4)	ADDRESS	4	TAS_DEFERRED_ABEND_R14_SAVE	Saved R14 when stack modified for deferred-abend.
(B8)	CHARACTER	4	TAS_DEFERRED_ABEND_CODE	Deferred abend code
(BC)	ADDRESS	4	TAS_NQ_ WORK_TOKEN	NQ work token
(C0)	CHARACTER	5	TAS_TCB_ID	tcb_id for trace
(C5)	CHARACTER	3	*	Reserved
(C8)	ADDRESS	4	* (4)	Reserved
(D8)	CHARACTER	256	TAS_PARAMETER_ LIST	Reply parameter list
(1D8)	CHARACTER	424	TAS_ERROR_INFORMATION	
(1D8)	CHARACTER	8	TAS_ERROR_CODE	Format: XXX/CCCC
(1E0)	UNSIGNED	1	TAS_ERROR_TYPE	Indicates the cause
(1E1)	BITSTRING	1	TAS_ERROR_ MVS_FLAGS	MVS Flags
	1...		TAS_ERROR_DUMP_REQUESTED	A dump was requested
	.111		TAS_ERROR_EXECUTING_RB	Flags determining error RB
	.1..		TAS_ERROR_SRB_MODE	Error in SRB mode
	..1.		TAS_ERROR_IRB	IRB on RB stack
	...1		TAS_ERROR_ CICS_RB_NOT_ACTIVE	CICS RB not in control
	.... 1...		*	Reserved
	.... .1..		TAS_ERROR_REASON_PRESENT	Abend reason code is present
	11		*	Reserved
(1E2)	BITSTRING	2	TAS_SYSTEM_INT	XXX (ie 00C1 for op exc)
(1E4)	BITSTRING	2	TAS_USER_INT	NNNN in binary
(1E6)	HALFWORD	2	TAS_ERROR_ OFFSET	Offset in program, or FFFF
(1E8)	CHARACTER	8	TAS_ERROR_ PROGRAM	Program in error
(1F0)	ADDRESS	4	TAS_ERROR_ ADDRESS	in error
(1F4)	FULLWORD	4	TAS_TAS_ATTACH_TOKEN	Attach token
(1F8)	ADDRESS	4	TAS_TAS_ TCA_ADDRESS	TCA address
(1FC)	ADDRESS	4	TAS_TAS_ADDRESS	Address of this task entry
(200)	FULLWORD	4	TAS_ERROR_ NUMBER	The number of this error
(204)	CHARACTER	4	TAS_ERROR_ REASON	Abend reason code
(208)	CHARACTER	160	TAS_CICS_DATA	Error data for CICS
(208)	CHARACTER	8	TAS_BC_PSW	
(210)	CHARACTER	8	TAS_EC_PSW	
(210)	CHARACTER	2	*	
(212)	BITSTRING	1	TAS_EC_BYTE3	
	1...		TAS_AR_MODE_ACTIVE	
(218)	CHARACTER	8	TAS_EC_ADD	
(220)	ADDRESS	4	TAS_INSTRUCTION_ADDRESS	
(224)	UNSIGNED	1	TAS_ERROR_KEY	
(225)	UNSIGNED	3	*	
(228)	CHARACTER	64	TAS_ERROR_REGISTER_STORAGE	
(228)	ADDRESS	4	TAS_ERROR_REGISTERS (16)	
(268)	CHARACTER	64	TAS_ERROR_ ACCESS_REG_STORAGE	
(268)	ADDRESS	4	TAS_ERROR_ACCESS_REGISTERS (16)	
(2A8)	CHARACTER		*	
(2A8)	CHARACTER	160	TAS_INT_DATA	
(2A8)	CHARACTER	8	TAS_BC_PSW	
(2B0)	CHARACTER	8	TAS_EC_PSW	
(2B0)	CHARACTER	2	*	

Offset Hex	Type	Len	Name (Dim)	Description
(2B2)	BITSTRING	1	TAS_EC_BYTE3	
	1...		TAS_AR_MODE_ACTIVE	
(2B8)	CHARACTER	8	TAS_EC_ADD	
(2C0)	ADDRESS	4	TAS_INSTRUCTION_ADDRESS	
(2C4)	UNSIGNED	1	TAS_ERROR_KEY	
(2C5)	UNSIGNED	3	*	
(2C8)	CHARACTER	64	TAS_ERROR_REGISTER_STORAGE	
(2C8)	ADDRESS	4	TAS_ERROR_REGISTERS (16)	
(308)	CHARACTER	64	TAS_ERROR_ ACCESS_REG_STORAGE	
(308)	ADDRESS	4	TAS_ERROR_ACCESS_REGISTERS (16)	
(348)	CHARACTER		*	
(348)	BITSTRING	8	TAS_ERROR_ TIMESTAMP	timestamp of error
(350)	CHARACTER	32	TAS_ERROR_ FP_REGS	FP register values:
(350)	CHARACTER	8	TAS_ERROR_ FP_REG_0	FP register 0
(358)	CHARACTER	8	TAS_ERROR_ FP_REG_2	FP register 2
(360)	CHARACTER	8	TAS_ERROR_ FP_REG_4	FP register 4
(368)	CHARACTER	8	TAS_ERROR_ FP_REG_6	FP register 6
The following two fields are only valid if TAS_ERROR_IN_SUBSPACE is set				
(370)	CHARACTER	8	TAS_ERROR_ STOKEN	Stoken for subspace
(378)	CHARACTER	4	TAS_ERROR_ALET	Alet for stoken
(37C)	BITSTRING	1	TAS_ERROR_SUBSPACE_FLAGS	
	1...		TAS_ERROR_IN_SUBSPACE	In a subspace?
	.111 1111		*	
(37D)	CHARACTER	3	*	Reserved
(380)	CHARACTER		*	Round to double-word

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	160	TAS_ERROR_DATA	
(0)	CHARACTER	8	TAS_BC_PSW	
(8)	CHARACTER	8	TAS_EC_PSW	
(8)	CHARACTER	2	*	Padding
(A)	BITSTRING	1	TAS_EC_BYTE3	
	1...		TAS_AR_ MODE_ACTIVE	AR_MODE FLAG
(10)	CHARACTER	8	TAS_EC_ADD	
(18)	ADDRESS	4	TAS_INSTRUCTION_ADDRESS	
(1C)	UNSIGNED	1	TAS_ERROR_KEY	TAS_EC_PSW key X'n0'
(1D)	UNSIGNED	3	*	Reserved
(20)	CHARACTER	64	TAS_ERROR_REGISTER_STORAGE	
(20)	ADDRESS	4	TAS_ERROR_ REGISTERS (16)	General Registers
(60)	CHARACTER	64	TAS_ERROR_ ACCESS_REG_STORAGE	
(60)	ADDRESS	4	TAS_ERROR_ACCESS_REGISTERS (16)	Access registers
(A0)	CHARACTER		*	Round to double-word

Error Table (including header)

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	21240	ERROR_TABLE	
(0)	CHARACTER	40	ERROR_HEADER	Error table header
(0)	CHARACTER	16	ERH_PREFIX	Standard prefix
(0)	HALFWORD	2	ERH_LENGTH	Length of table header
(2)	CHARACTER	1	ERH_ARROW	>
(3)	CHARACTER	3	ERH_DFH	DFH

Offset Hex	Type	Len	Name (Dim)	Description
(6)	CHARACTER	2	ERH_DOMID	KE
(8)	CHARACTER	8	ERH_BLOCK_NAME	ERH
(10)	ADDRESS	4	ERH_TABLE_START	First error table entry
(14)	ADDRESS	4	ERH_TABLE_END	End of error table
(18)	HALFWORD	2	ERH_ENTRY_ LENGTH	Length error table entry
(1A)	HALFWORD	2	*	Reserved
(1C)	FULLWORD	4	*	Reserved
(20)	CHARACTER	8	ERH_QUICK_CELL	
(20)	FULLWORD	4	ERH_FIRST_FREE	Index of next free entry (1..ERROR_ENTRY_NUMBER)
(24)	FULLWORD	4	ERH_GUARD	Quick-cell guard count = number of errors so far
(28)	CHARACTER	424	ERROR_ENTRY (50)	Error table entries
(52F8)	CHARACTER		*	Round to double-word

KTCB Table Header

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	KTCB_HEADER	KTCB table header
(0)	CHARACTER	16	KTCH_PREFIX	Standard prefix
(0)	HALFWORD	2	KTCH_LENGTH	Length of table header
(2)	CHARACTER	1	KTCH_ARROW	>
(3)	CHARACTER	3	KTCH_DFH	DFH
(6)	CHARACTER	2	KTCH_DOMID	KE
(8)	CHARACTER	8	KTCH_BLOCK_NAME	KTCH
(10)	ADDRESS	4	KTCH_TABLE_START	First KTCB table entry
(14)	ADDRESS	4	KTCH_LAST_ENTRY	Last KTCB table entry
(18)	HALFWORD	2	KTCH_ENTRY_ LENGTH	Length of KTCB table entry
(1A)	HALFWORD	2	*	Reserved
(1C)	CHARACTER	4	*	Reserved
(20)	CHARACTER	16	KTCH_SPECIFIC_ TCBS	Named KTCB table entries
(20)	ADDRESS	4	KTCH_STEP_TCB	-> Job Step TCB entry
(24)	ADDRESS	4	KTCH_FO_TCB	-> File Owning TCB
(28)	ADDRESS	4	KTCH_RO_TCB	-> Resource Owning TCB
(2C)	ADDRESS	4	KTCH_QR_TCB	-> Quasi Re-entrant TCB
(30)	CHARACTER	8	KTCH_QUICK_CELL	
(30)	ADDRESS	4	KTCH_FIRST_FREE	First KTCB in free list
(34)	FULLWORD	4	KTCH_GUARD	Quick-cell guard count
(38)	CHARACTER		*	Round to double-word

KTCB Table Entry

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4064	KTCB_ENTRY	KTCB table entry
(0)	CHARACTER	8	KTCB_NAME	Eye-catcher KTCB
(8)	ADDRESS	4	KTCB_NEXT_FREE	Free list pointer
(C)	ADDRESS	4	KTCB_DEFAULT_ TASK	Default task for this TCB

NB. Next field (KTCB_ACTIVE_TASK) is also declared in DFHKEPRP
for user usage via DFHKERN, and it MUST BE KEPT IN SYNC

Offset Hex	Type	Len	Name (Dim)	Description
(10)	ADDRESS	4	KTCB_ACTIVE_TASK	Task this TCB is executing
(14)	ADDRESS	4	KTCB_STEAL_POINT	Address of stack entry to steal from
(18)	CHARACTER	24	KTCB_TIMER	Timer management fields
(18)	CHARACTER	8	KTCB_ACCUM_TIME	Accumulated TCB time
(20)	CHARACTER	8	KTCB_STIMER_ TIME	Time last STIMER was issued
(28)	CHARACTER	8	KTCB_EXIT_TIME	Value last STIMER interval
(30)	CHARACTER	4	KTCB_TIMER_STATE	Status of CPU timing
	1...		KTCB_TIMER_ ACTIVE	CPU timing is active
(30)	BITSTRING	1	*	Reserved
(32)	HALFWORD	2	KTCB_TIMER_ CHANGES	Number of times state has changed
(34)	FULLWORD	4	KTCB_TCB_ WAIT_ECB	ECB used to Wait this TCB for Perform_System_Action
(38)	BITSTRING	2	KTCB_STATE	Status of TCB
	1...		KTCB_SWITCH_ SS_ENV	Switch ENVIRONMENT
	.1..		KTCB_SS_ENV	SUBSPACE ENVIRONMENT
	..1.		KTCB_LE_CICS	LE uses CICS services
	...1		KTCB_EXEC_ CAPABLE	supports EXEC CICS
	.... 1...		KTCB_UNUSED	KTCB entry not in use
	.... .1..		KTCB_ATTACHED_ TCB	TCB is attached-unlike Step
	.... ..1.		KTCB_CURRENTLY_ ATTACHED	TCB is currently attached
	1		KTCB_TCB_POSTED	MVS Posted for termination
(39)	1...		KTCB_ESSENTIAL_ TCB	

Offset Hex	Type	Len	Name (Dim)	Description
				essential TCB - '1'b
	.1..		KTCB_DAUGHTER_TERMINATED	Daughter can be detached. *
	..1.		KTCB_HAS_BEEN_DETACHED	Corr TCB has been detached *
	...1		KTCB_ATTACHING_ TCB	TCB IS being attached.
	.... 1...		KTCB_ESTAE_ENVIRONMENT	TCB IS to be terminated. *
	.... .111		*	
(3A)	BITSTRING	1	KTCB_ESTAE_STATE	Status of Estae
	1...		KTCB_KESTX_IN_PROGRESS	DFHKESTX is in control
	.1..		*	Reserved
	..1.		KTCB_CLEAN_ UP_ESTAE	SDWACLUP was set
	...1		KTCB_CANCEL_ ESTAE	X22 Abend (Cancel)
	.... 1...		KTCB_NO_SDWA	No SDWA for DFHKESTX
(3B)	BITSTRING	1	KTCB_ABEND_999	Type of Abend 999 request
	1...		KTCB_RUNAWAY_REQUESTED	Abend 999 runaway request
	.1..		KTCB_RESET_REQUESTED	Abend 999 reset PSW request
	..1.		KTCB_PERCOLATE_ERROR	Abend 999 percolate error
	...1		KTCB_OUT_ OF_STACK	Abend 999 out of stack
	.... 1...		KTCB_ERROR_MAX_EXCEEDED	ABEND 999 MAX ERR
	.... .111		*	Reserved
(3C)	CHARACTER	1	KTCB_TCB_TYPE	TCB type: S - Job step R - Resource owning Q - Quasi re-entrant C - Concurrent Z - Secondary LU P - ONC/RPC N - modename
(3D)	CHARACTER	2	KTCB_MODENAME	TCB modename:
(40)	ADDRESS	4	KTCB_TRAP_ PARAMETER	Global trap parameter list
(48)	CHARACTER	20	KTCB_ATTACH_INTERFACE	Interface to MVS Attach
(48)	ADDRESS	4	KTCB_ATTACH_ PARAM	Address of the TCB entry
(4C)	FULLWORD	4	KTCB_ATTACH_ INIT_ECB	This ECB is Posted when Create TCB selects this TCB
(50)	ADDRESS	4	KTCB_ATTACH_TCB_ADDRESS	Address of MVS TCB for this KTCB entry
(54)	FULLWORD	4	KTCB_TERMINATE_ ECB	This ECB is Posted to force the Step TCB to terminate
(58)	ADDRESS	4	KTCB_MVS_RSA	MVS save area passed from MVS by the newly Attached TCB
(5C)	ADDRESS	4	KTCB_RESET_PARAMETER	PSW and registers for Reset
(60)	CHARACTER	20	KTCB_LOCK_ ELEMENT	TCB lock queue element
(60)	CHARACTER	8	KTCB_LOCK_ STATIC_QEL	CHAR(8)
(60)	FULLWORD	4	*	
(64)	ADDRESS	4	KTCB_LOCK_ CHAIN	Next TCB lock queue element *
(68)	ADDRESS	4	KTCB_LOCK_BACK_POINTER	Lock block address
(68)	ADDRESS	4	KTCB_LOCK_ LCB_PTR	
(6C)	ADDRESS	4	KTCB_LOCK_ACTIVE_QEL_PTR	
(70)	FULLWORD	4	KTCB_LOCK_ECB	ECB used to wait this TCB *
(74)	CHARACTER	16	KTCB_TCB_TOKEN	
(84)	ADDRESS	4	KTCB_RESET_ FP_REGS	FP registers for Reset
(88)	ADDRESS	4	KTCB_NEXT_ENTRY	Next table entry
(8C)	ADDRESS	4	KTCB_MOTHER_KTCB	Address of mother KTCB
(90)	HALFWORD	2	KTCB_PRTY_ RELATIVE_TO_PARENT	
(92)	BITSTRING	1	KTCB_CANCEL_ STATE	Status of CANCeL
	1...		KTCB_CANCEL_REQUESTED	ABEND 999 CANCEL REQD
	.111 1111		*	Reserved
(98)	CHARACTER	8	KTCB_KETIX_LAST_INVOKED	Time of last KETIX run

The following four fields are used as automatic storage for new variables to one of these modules.

Offset Hex	Type	Len	Name (Dim)	Description
(A0)	CHARACTER	2700	KTCB_ESTAE_ AUTOMATIC	Auto for Estae exit
(B30)	CHARACTER	336	KTCB_STIMER_ AUTOMATIC	Automatic for Stimer exit
(C80)	CHARACTER	64	KTCB_ETXR_ AUTOMATIC	Automatic for ETXRer exit
(CC0)	CHARACTER	800	KTCB_TCB_ AUTOMATIC	Automatic for TCB code
(FE0)	CHARACTER		KTCB_AUTOMATIC_ END	End of automatic areas
(FE0)	CHARACTER		*	Round to double-word

Constants

Len	Type	Value	Name	Description
0	BIT	000	TAS_ERROR_CICS_RB	
Possible values for KTCB_ TCB_TYPE				
1	CHARACTER	S	KTCB_JOB_STEP	
1	CHARACTER	F	KTCB_FILE_OWNING	
1	CHARACTER	R	KTCB_RESOURCE_ OWNING	
1	CHARACTER	Q	KTCB_QUASI_ REENTRANT	
1	CHARACTER	C	KTCB_CONCURRENT	
1	CHARACTER	Z	KTCB_SECONDARY_LU	
1	CHARACTER	P	KTCB_ONC_RPC	
1	CHARACTER	N	KTCB_ARBITRARY_ NAME	
Error Table Constant				
4	DECIMAL	50	ERROR_ENTRY_NUMBER	

KEMHD Kernel module header

CONTROL BLOCK NAME = DFHKEMHD
DESCRIPTIVE NAME = CICS (KE) Module header
FUNCTION =
Define the module header control block.
LIFETIME =
Same as the module which contains the module header.
STORAGE CLASS =
Same as the module which contains the module header.
LOCATION =
At the start of any module which contains the module header.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES =
DATA AREAS =
CONTROL BLOCKS =
GLOBAL VARIABLES (Macro pass) =

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	72	MODULE_DESCRIPTOR	
(0)	HALFWORD	2	MODHHLEN	THIS DSECT LENGTH
(2)	CHARACTER	8	MODHEYE	Eyecatcher '>MODHEAD' *
(A)	UNSIGNED	1	MODHLEVL	LEVEL = 03
(B)	CHARACTER	1	MODHLANG	LANG A=ASM P= PLS
(C)	CHARACTER	1	MODHSYST	ATTRIBUTE ONE
	1...		MODHOS	MVS
	.1..		MODHDOS	DOS
	..1.		MODHCMS	CMS
	...1 1111		*	...
(D)	CHARACTER	3	MODHRELS	RELEASE OF CICS
(10)	CHARACTER	8	MODHNAME	FULL NAME
(18)	CHARACTER	8	MODHDATE	DATE OF ASSEMBLY
(20)	CHARACTER	1	*	
(21)	CHARACTER	5	MODHTIME	TIME OF ASSEMBLY
(26)	UNSIGNED	1	MODHATR1	ATTRIBUTE ONE
(27)	BITSTRING	1	MODHATR2	ATTRIBUTE BYTE TWO
	1111 1...		*	For Future Use.
	.... .1..		MODH_AUTOREG_13	1 = autoreg_13, 0 = not
	.... ..1.		MODH_HANDLE_ DEF_ABEND	

Offset Hex	Type	Len	Name (Dim)	Description
				1 = handles deferred abend, 0 = doesn't
	1		MODHAM31	Amode. 0 = 24, 1 = 31.
(28)	ADDRESS	4	MODHRCVR	Address of recovery routine
(2C)	CHARACTER	8	MODHSERV	Service Data (PTF/APAR)
(34)	CHARACTER	4	MODHIPROC	IPROC Data.
(34)	HALFWORD	2	MODH_IPROC_D	IPROC Descriptor: Offset in module.
(36)	HALFWORD	2	MODH_IPROC_F	IPROC Flags: Offset in automatic.
(38)	UNSIGNED	2	MODHSOFF	Offset to static
(3A)	UNSIGNED	1	MODHSNUM	Num. of static regs
(3B)	UNSIGNED	1	MODHCNUM	Number of Code Registers
(3C)	HALFWORD	2	*	For future use.
(3E)	UNSIGNED	2	MODHMLEN	MODULE LENGTH
(40)	FULLWORD	4	MODHSTKL	REQUIRED STACK LENGTH
(44)	FULLWORD	4	MODHSMODE	Smode index

Lifo Plist

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	17	DFHLIFO_PLIST	Lifo Plist.
(0)	HALFWORD	2	LF_PLIST_LEN	Length of Plist.
(2)	HALFWORD	2	LF_PLIST_DID	DSA Id.
(4)	HALFWORD	2	LF_PLIST_DLN	DSA Length.
(6)	HALFWORD	2	LF_PLIST_MODULE_OFFSET	Offset of Module Start from where this Plist is.
(8)	FULLWORD	4	LF_PLIST_TRC	Trace Flags.
(C)	HALFWORD	2	LF_PLIST_MOD	Module Id.
(E)	CHARACTER	2	LF_PLIST_MDC	Module Id in Character form.
(10)	BITSTRING	1	LF_PLIST_TRF	Option Setting.
	1111		*	Padding.
	.... 1...		LF_PLIST_TRCN	Conditional Request.
	.... .1..		LF_PLIST_TRRN	Conditional Return Request.
	.... ..1.		LF_PLIST_TRIC	IC Logic is requested.
	1		LF_PLIST_TRTR	Tracing is requested.

Constants

Len	Type	Value	Name	Description
8	CHARACTER	>MODHEAD	MODH_EYE_CATCHER	
EQUATES FOR MODHATR1.				
1	DECIMAL	0	MODHATRD	READONLY
1	DECIMAL	1	MODHATNR	NON READONLY
1	DECIMAL	2	MODHATRE	FULLY REENTRANT
Equates for MODHSMODE.				
4	DECIMAL	0	MODHSMODE_31	Smode 31
4	DECIMAL	8	MODHSMODE_24	Smode 24

KESTP Kernel stack entry

CONTROL BLOCK NAME = DFHKESTP
DESCRIPTIVE NAME = CICS (KE) Kernel Stack Structure.
FUNCTION =
LIFETIME = Per Call.
STORAGE CLASS = Kernel-Managed MVS Storage /
KESTACKS subpool storage
LOCATION = R13 -> this block.
INNER CONTROL BLOCKS =
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
Kernel Stack
Format must remain compatible with LIFO stack.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	128	KERNSTCK	
(0)	CHARACTER	76	*	
(0)	CHARACTER	1	KERNOFF0	Type of stack entry
(1)	UNSIGNED	1	KERNSTAT	Status flags
	1...		KERNLOOP	DSA may be looping
	.1..		KERNERRD	DFHKERRD exists, i.e. stack in error state
	..1.		KERNACR	CICS Recovery added
	...1		KERNSAVE	Save area exists and is pointed to by KERNSAVP
	.... 1...		KERNLCON	Loop controller
	.... .1..		KERNDFAB	Deferred abend scheduled against this stack
(2)	HALFWORD	2	KERNOFLN	Length of stack+auto
(4)	ADDRESS	4	KERNBPTR	Backward stack pointer
(8)	ADDRESS	4	*	Reserved
(C)	CHARACTER	64	KERNRGST	Registers 14:13
(C)	ADDRESS	4	KERNREGS (16)	Registers 14:13 R1 = Address of plist
(4C)	ADDRESS	4	KERNSAVP	Save area pointer
(50)	ADDRESS	4	KERNTASN	Address ot task entry
(54)	ADDRESS	4	KERNPOWN	Address of kernel global storage
(58)	ADDRESS	4	KERNDTAB	Caller's domain entry
(5C)	BITSTRING	4	KERNTRFL	Trace flags(1 = trace)
(60)	ADDRESS	4	KERNNAB	Next available byte
(64)	ADDRESS	4	KERNMODH	header
(68)	FULLWORD	4	KERNSGCN	Segment chain DSA back chai
(6C)	ADDRESS	4	*	Reserved
(70)	CHARACTER	4	KERNMODS	Module name IDs
(70)	ADDRESS	4	KERNSCCN	Saved Lifo back chain (Subroutine call/retn only)
(74)	ADDRESS	4	*	Reserved.
(78)	ADDRESS	4	*	Reserved.
(7C)	ADDRESS	4	*	Reserved.
(80)	CHARACTER		KERNSTCK_END	Round to double-word - See note above about changing the length of this structure.

Kernel Stack Save Area

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	KESTACKSAVE	
(0)	CHARACTER	196	KES_HEADER	
(0)	CHARACTER	128	KES_SAVED_ STACK_ENTRY	Saved stack entry
(80)	CHARACTER	64	KES_REGISTERS	Register save area
(C0)	FULLWORD	4	KES_LENGTH	Incl. length of save area *
(C4)	CHARACTER	*	KES_AUTOMATIC	Automatic storage

Kernel Domain Table Entry Overlay.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	KERN_DTE	
(0)	CHARACTER	8	*	Used by Kernel
(8)	FULLWORD	4	KERN_DTE_INDEX	Domain index

Offset Hex	Type	Len	Name (Dim)	Description
(C)	CHARACTER	4	*	USED BY KERNEL
(10)	ADDRESS	4	KERN_DTE_ANCHOR	Domain anchor
(14)	CHARACTER	*	*	Used by Kernel

Constants

Len	Type	Value	Name	Description
1	CHARACTER	9	KERN0KER	
1	DECIMAL	0	KERN0DCL	
1	CHARACTER	1	KERN0SCL	
1	CHARACTER	2	KERN0LCL	

LDCBS Loader domain control blocks

```
Segment Name = DFHLDCBS
DESCRIPTIVE NAME = CICS Loader (LD) Domain
              Control Block declarations.
Function =
    This file contains the control block and constant
    declarations used by the Loader domain.
    The file is included by each Loader domain
    module.
    The control blocks are:
        APE - Active Program Element.
        BLDL - BLDL PARAMETER LIST.
        CPE - Current Program Element.
        CSECTL - CSECT LIST BLOCK AND ENTRY.
        DUMMY_CDE - used by SLD
        DUMMY_XTLST - used by SLD
        DUMP - LOADER DUMP CODES.
        GLOBAL - Loader global storage area.
        LAFPB - LOADER AUTHORISED FACILITIES PARAMETER BLOCK.
        LDBE - Loader Domain Browse Element.
        LDWE - Loader Domain Wait Element.
        LOB - LOADER OPTION BLOCK.
        MSGS - LOADER MESSAGE NUMBERS.
        PDB - Program Descriptor Block.
        TRACE - Trace point definitions.
    Each control block declaration is followed by the
    constant declarations related to it.
Notes:
 Dependencies = S/370
 Restrictions = none
 Register Conventions = domain standard (no special usage)
 Patch Label = N/A
 Module Type = N/A
 Attributes = N/A
A P E - ACTIVE PROGRAM ELEMENT
-----
For each instance of a program currently loaded there will be a
associated APE. A program instance is associated with it's
definition by chaining the APE to the CPE (Current Program
Element).
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	136	APE	
(0)	CHARACTER	48	APE_PREFIX	
(0)	UNSIGNED	2	APE_LENGTH	APE control block length
(2)	CHARACTER	1	APE_ARROW	Control Block eyecatcher
(3)	CHARACTER	3	APE_DFH	
(6)	CHARACTER	2	APE_DOMAIN	
(8)	CHARACTER	8	APE_BLOCK_ID	
(10)	CHARACTER	8	APE_PROGRAM_ NAME	Program name

APE Chain Fields, there are three APE chains:

1. The Global APE chain which is anchored in LD Global. This contains all the APEs in the system in ascending order of entry point.
2. The CPE/APE chain which is anchored in the CPE. This chains all the instances of a program to the program definition. The most recently acquired instance is at the head of the chain.
3. The APE Not-In-Use (NIU) chain which is anchored in LD Global This contains all the APEs in the system which have a use count of zero and are defined as REUSABLE. During program compression these programs are eligible to be removed. APEs are added to the head of the chain and only removed if the program is freemained or reused.

Offset Hex	Type	Len	Name (Dim)	Description
(18)	CHARACTER	24	APE_CHAIN_ FIELDS	APE chain fields.
(18)	ADDRESS	4	APE_NEXT	-> next APE in Global APE chain.
(1C)	ADDRESS	4	APE_PRIOR	-> previous APE in Global APE chain.
(20)	ADDRESS	4	APE_OLDER_APE	-> older APE in CPE/ APE chain
(24)	ADDRESS	4	APE_YOUNGER_ APE	-> younger APE in CPE /APE chain
(28)	ADDRESS	4	APE_OLDER_ APE_NIU	-> older APE in APE NIU chain.
(2C)	ADDRESS	4	APE_YOUNGER_ APE_NIU	-> younger APE in APE NIU chain.
(30)	ADDRESS	4	APE_OWNING_CPE	Address of owning CPE

The Program Descriptor Block (PDB) is copied into the APE.

Offset Hex	Type	Len	Name (Dim)	Description
(34)	CHARACTER	16	APE_PDB	Prog Descriptor flds
(44)	UNSIGNED	1	APE_STATUS	Status: active/freed

Attributes of the program associated with this APE.

Offset Hex	Type	Len	Name (Dim)	Description
(45)	UNSIGNED	1	APE_FLAGS	Attributes of program instance
	1...		APE_LPA_LOADED	Program LPA resident
	.1..		APE_RPL_LOADED	Program RPL loaded
	..1.		APE_REGION_ LOADED	Program region loaded
	...1		APE_RMODE_ANY	Program RMODE ANY
	.... 1...		APE_MUSTDELET	= PMARL_MUSTDELET
	.... .1..		*	Reserved
	.... ..1.		APE_AMODE_31	Program AMODE 31
	1		APE_AMODE_24	Program AMODE 24
(46)	UNSIGNED	1	APE_RECOVERY_ FLAGS	
	1...		APE_BUILT_ BY_RESTART	Prog loaded during init.
	.111 1111		*	Reserved
(47)	UNSIGNED	1	*	Reserved
(48)	FULLWORD	4	APE_COPY_NUMBER	Copy no. of the APE
(4C)	FULLWORD	4	APE_LOAD_POINT	Load point of program
(50)	FULLWORD	4	APE_ENTRY_POINT	Entry point of program
(54)	FULLWORD	4	APE_PROGRAM_ LENGTH	length of program
(58)	FULLWORD	4	APE_CURRENT_ USERS	No. of users
(5C)	FULLWORD	4	APE_STORAGE_SIZE	storage allocated to prog.
(60)	CHARACTER	12	APE_SUBPOOL_DATA	Subpool prog. was getmained from
(60)	CHARACTER	8	TOKEN	
(68)	UNSIGNED	4	DSA	
(6C)	FULLWORD	4	APE_CSECT_ LIST_SIZE	No. of CSECT list blocks chained to this APE.
(70)	CHARACTER	8	APE_CSECT_ LIST_CHAIN_FIELDS	Next and prior ptrs
(78)	CHARACTER	8	APE_ON_NIU_TIME	Time APE put on NIU chain
(80)	ADDRESS	4	APE_DUMMY_CDE	-> to dummy CDE

If APE_MUSTDELET is set, delete needs the loader token ...

Offset Hex	Type	Len	Name (Dim)	Description
(84)	FULLWORD	4	APE_BLITO	offset within program
(88)	CHARACTER		*	

BLDL_ LIST - BLDL PARAMETER LIST
The BPAM directory entry is built by the MVS LLACOPY interface and contains a copy of the directory entry from the Partitioned Dataset (PDS) containing the named program.
The BLDL parameter list passed on the LLACOPY is a series of directory entries preceded by entry count and entry length fields.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	22	BLDL_LIST	
(0)	CHARACTER	18	BLDL_PREFIX	Control block eyecatcher
(0)	FULLWORD	4	BLDL_LENGTH	Control block length
(4)	CHARACTER	1	BLDL_ARROW	
(5)	CHARACTER	3	BLDL_DFH	

Offset Hex	Type	Len	Name (Dim)	Description
(8)	CHARACTER	2	BLDL_DOMAIN	
(A)	CHARACTER	8	BLDL_BLOCK_ID	

The BLDL macro parameter list

Offset Hex	Type	Len	Name (Dim)	Description
(12)	CHARACTER	4	BLDL_MACRO_PLIST	
(12)	UNSIGNED	2	BLDL_NUMBER_ IN_LIST	No of entries in list
(14)	UNSIGNED	2	BLDL_LENGTH_OF_ENTRY	Length of BLDL list
(16)	CHARACTER		BLDL_ENTRIES	The BLDL entries

The BLDL_LIST_ENTRY is a duplicate of the PDS entry for the program, hence, do not attempt to use any of the reserved fields.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	60	BLDL_LIST_ENTRY	BLDL list entry
(0)	CHARACTER	8	BLDL_PROGRAM_ NAME	Program name
(8)	UNSIGNED	4	BLDL_TTRK	Track and record data
(8)	CHARACTER	2	BLDL_TT	Relative track
(A)	UNSIGNED	1	BLDL_R	Relative record
(B)	UNSIGNED	1	BLDL_LCN	Concatenation No. of dataset
(C)	UNSIGNED	1	BLDL_WHERE_FOUND	Library flag field
(D)	UNSIGNED	1	BLDL_C_FIELD	Indicator byte
	1...		BLDL_ALIAS	Name is an alias
	.11.		*	Reserved
	...1 1111		*	Reserved
(E)	CHARACTER	8	*	
(16)	UNSIGNED	1	BLDL_ATTRIBUTE	Program attributes
	1111 11..		*	Reserved
	.... ..1.		BLDL_EXECUTABLE	Program executable
	1		*	Reserved
(17)	CHARACTER	1	*	Reserved
(18)	UNSIGNED	3	BLDL_PROGRAM_ LENGTH	Program length
(1B)	CHARACTER	2	*	Reserved
(1D)	UNSIGNED	3	BLDL_ENTRY_POINT_OFFSET	Entry point offset
(20)	CHARACTER	1	*	Reserved
(21)	UNSIGNED	1	BLDL_FLAGS_2	
	111.		*	Reserved
	...1		BLDL_RMODE_ANY	'1' RMODE ANY '0' RMODE 24
	.... 11..		*	Reserved
	.... ..1.		BLDL_AMODE_31	'1' AMODE 31 '0' AMODE 24
	1		*	Reserved
(22)	CHARACTER	26	*	Reserved

C P E - CURRENT PROGRAM ELEMENT

A Current Program Element represents a program defined to Loader.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	176	CPE	
(0)	CHARACTER	24	CPE_PREFIX	Standard prefix
(0)	UNSIGNED	2	CPE_LENGTH	Control block length
(2)	CHARACTER	1	CPE_ARROW	Control block eyecatcher
(3)	CHARACTER	3	CPE_DFH	
(6)	CHARACTER	10	CPE_EYE_CATCH	
(6)	CHARACTER	2	CPE_DOMAIN	
(8)	CHARACTER	8	CPE_BLOCK_ID	

CPE chain is anchored in LD Global. It contains all the CPEs (programs currently defined to the system) stored in alphabetical order by program name.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	ADDRESS	4	CPE_NEXT	-> next CPE in chain
(14)	ADDRESS	4	CPE_PRIOR	-> previous CPE in chain

The following are valid CPE statuses:

UNUSED - program has been defined but not yet acquired.

LOCATED - An LLACOPY has been issued for the program and it has been found in the DFHRPL library.

LOADED - The program has been loaded. It should be noted that a CPE defined as RELOAD will never have the status updated to loaded, hence, on every acquire a new program instance is loaded. Also, if a REFRESH PROGRAM is requested (CEMT S NEWCOPY) the status will be reduced to LOCATED for a DFHRPL loaded program.

DELETED - The program definition has been deleted ie DELETE_ PROGRAM has been issued. The CPE has not been freemained as there are still active APEs chained off it. The CPE will be freemained when all the active APEs are released.

BAD - Invalid data has been detected in the CPE, hence, it is marked as unusable.

FREED - The CPE has been freemained. This status is solely to mark deleted CPEs in the case where they are not overwritten and they appear in a dump.

Offset Hex	Type	Len	Name (Dim)	Description
(18)	UNSIGNED	1	CPE_PROGRAM_ STATUS	Status of the program

The CPE control block lock is used to ensure that it is not possible to have multiple updates of a CPE. While a CPE is locked no other task may access it.

UNLOCKED - No task is currently attempting to update the CPE.

LPA_LOCATING - A task is currently attempting to locate a program in the LPA.

RPL_LOCATING - A task is currently attempting to locate a program in the DFHRPL library.

RPL_LOADING - A task is currently attempting to load a program from thr DFHRPL library.

DISCONNECTING - A task is currently disconnecting from the member in RPL

(19)	UNSIGNED	1	CPE_LOCK	CPE lock field
(1A)	UNSIGNED	1	CPE_RECOVERY_ FLAGS	
	1...		CPE_BUILT_ BY_RESTART	CPE built during init.
	.1..		CPE_LOADED_ BY_RESTART	Program loaded during init
	..1.		CPE_PRVMOD	Program should be loaded from RPL even though it is resident in the LPA
	...1		CPE_PROGRAM_ ACQUIRED	program loaded and has been ACQUIREd
	.... 1...		CPE_OLD_ COPY_IN_LPA	Program has already been defined and is resident in the LPA.
	.... .1..		CPE_PMARL_VALID	PMARL has been fetched@LEA
	.... ..1.		CPE_MUSTDELET	= PMARL_MUSTDELET
	1		*	Reserved
(1B)	UNSIGNED	1	CPE_PDB_ CATALOG_STATUS	Shows if PDB has been cataloged

The CPE_DE (directory entry) is copied from the BLDL_LIST_ENTRY ,obtained when the LLACOPY is issued for the program. For details of the fields see the BLDL_LIST_ENTRY.

(1C)	CHARACTER	60	CPE_DE	CPE directory entry
(1C)	CHARACTER	8	CPE_PROGRAM_ NAME	
(24)	UNSIGNED	4	CPE_TTRK	
(24)	UNSIGNED	2	CPE_TT	
(26)	UNSIGNED	1	CPE_R	
(27)	UNSIGNED	1	CPE_LCN	
(28)	UNSIGNED	1	CPE_Z_BYTE	
(29)	UNSIGNED	1	CPE_C_BYTE	
(2A)	CHARACTER	8	*	
(32)	UNSIGNED	1	CPE_ATTRIBUTES	
	1...		CPE_REENTRANT	
	.111 1111		*	
(33)	CHARACTER	1	*	
(34)	UNSIGNED	3	CPE_PROGRAM_ LENGTH	
(37)	CHARACTER	2	*	
(39)	UNSIGNED	3	CPE_ENTRY_ POINT_OFFSET	
(3C)	CHARACTER	1	*	
(3D)	UNSIGNED	1	CPE_FLAGS	
	111.		*	
	...1		CPE_RMODE_ANY	
	.... 11..		*	
	.... ..1.		CPE_AMODE_31	
	1		*	
(3E)	CHARACTER	26	*	

Offset Hex	Type	Len	Name (Dim)	Description
The Program Descriptor Block (PDB) is copied in here.				
(58)	CHARACTER	16	CPE_PDB	
CPE statistics				
(68)	FULLWORD	4	CPE_USES	Cummulative count of the no. of times a program is acquired.
(6C)	FULLWORD	4	CPE_CURRENT_ USERS	No. of current users.
(70)	FULLWORD	4	CPE_LOAD_COUNT	No. of times a program has been loaded
CPE APE chain This chain contains an APE for each instance of THIS program currently in main storage. New APEs are added to the head of chain.				
(74)	FULLWORD	4	CPE_APE_ CHAIN_SIZE	No. of APEs currently chained to this CPE
(78)	CHARACTER	24	CPE_APE_ CHAIN_FIELDS	
CPE statistics These figures are the official statistics and are reset at the end of a statistics collection.				
(90)	CHARACTER	24	CPE_STATS	
(90)	FULLWORD	4	CPE_TIMES_USED	Cummulative count of the no. of times a program is acquired.
(94)	FULLWORD	4	CPE_FETCH_COUNT	No. of times a program has been loaded from the RPL or located in the LPA.
(98)	FULLWORD	4	CPE_LOAD_TIME	Cummulative load duration for all MVS loads.
(9C)	FULLWORD	4	CPE_COMPRESSIONS	No. of times a copy of this program has been removed due to proram compression
(A0)	FULLWORD	4	CPE_WAITS	No. of times tasks were forced to wait due to the CPE being locked.
(A4)	FULLWORD	4	CPE_REFRESHES	No. of times the program has been refreshed.
(A8)	ADDRESS	4	CPE_GLOB_PTR	-> back to global
(AC)	FULLWORD	4	CPE_BLITO	Offset to IEWBLIT
(B0)	CHARACTER		*	

CESECTL - CSECT LIST
The CESCT list contain the CSECT name ,the address of the CSECT, the CICS version, the PTF level and time the CSECT was last updated. A CSECTL only contains four entries, therefore, several CSECTL blocks maybe chained off the APE. The CSECT details are obtained from the header data.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	176	CSECTL	
(0)	CHARACTER	24	CSECTL_PREFIX	Control block prefix
(0)	UNSIGNED	2	CSECTL_LENGTH	Control block length
(2)	CHARACTER	1	CSECTL_ARROW	Control block eyecatcher
(3)	CHARACTER	3	CSECTL_DFH	
(6)	CHARACTER	2	CSECTL_DOMAIN	
(8)	CHARACTER	8	CSECTL_BLOCK_ID	
(10)	CHARACTER	8	CSECTL_ CHAIN_FIELDS	CSECTL chain fields anchored in the associated APE
(10)	ADDRESS	4	CSECTL_NEXT	->to next CSECTL block
(14)	ADDRESS	4	CSECTL_PRIOR	->to previous CSECTL block
CSECTL list entries.				
(18)	CHARACTER	38	CSECTL_ENTRIES (4)	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	38	CSECTL_ENTRY	
(0)	CHARACTER	8	CSECTL_MODULE	CSECT name
(8)	ADDRESS	4	CSECTL_ADDRESS	Address of CSECT
(C)	CHARACTER	4	CSECTL_ CICS_VERSION	CICS version
(10)	CHARACTER	8	CSECTL_PTF_LEVEL	PTF level of CSECT
(18)	CHARACTER	14	CSECTL_CREATION	Time CSECT last updated

CDE - DUMMY CDE

The DUMMY CDE is used by SLD to detect mdules loaded by the CICS Loader. As the MVS LOADs are directed no CDE is built so we have to supply a dummy one so SLD can set its breakpoints.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	DUMMY_CDE	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER	24	DUMMY_CDE_PREFIX	
(0)	UNSIGNED	2	DUMMY_CDE_ LENGTH	
(2)	CHARACTER	1	DUMMY_CDE_ARROW	
(3)	CHARACTER	3	DUMMY_CDE_DFH	
(6)	CHARACTER	2	DUMMY_CDE_ DOMAIN	
(8)	CHARACTER	8	DUMMY_CDE_ BLOCK_ID	
(10)	CHARACTER	8	DUMMY_CDE_CHAIN	
(10)	ADDRESS	4	DUMMY_CDE_NEXT	
(14)	ADDRESS	4	DUMMY_CDE_PREV	

The following must be kept in step with the IHACDE DSECT SLD
should only check CDCHAIN, CDNAME, CDENTPT AND CDXLMJP.

Offset Hex	Type	Len	Name (Dim)	Description
(18)	CHARACTER	32	DUMMY_CDE_ CONTENTS	
(18)	ADDRESS	4	DUMMY_CDCHAIN	-> next CDE
(1C)	ADDRESS	4	*	Reserved
(20)	CHARACTER	8	DUMMY_CDNAME	Name
(28)	FULLWORD	4	DUMMY_CDENTPT	Entry point top bit set for amode
(2C)	ADDRESS	4	DUMMY_CDXLMJP	-> extent list (XTLST)
(30)	CHARACTER	8	*	Reserved

X T L S T - Dummy Extent List

The DUMMY XTLST is used by SLD to detect modules loaded by the CICS Loader. As the MVS LOADs are directed no CDE or extent lists are built so we have to supply dummy ones so SLD can set its breakpoints.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	DUMMY_XTLST	

The following must be kept in step with the IHAXTLST DSECT
SLD should only check XTLMSBLA and XTLMSBAA.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHARACTER	8	*	Reserved
(8)	FULLWORD	4	DUMMY_XTLMSBLA	Pgm length
(C)	ADDRESS	4	DUMMY_XTLMSBAA	Load point

G L O B A L - LOADER GLOBAL AREA
The Loader Global area (anchor block) contains the domain status indicator, storage subpool tokens, lock tokens, CPE chain anchor, APE chain anchor, APE NIU chain anchor and the statistics buffer anchor.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	912	GLOBAL	
(0)	UNSIGNED	2	GLOBAL_LENGTH	Control block length
(2)	CHARACTER	1	GLOBAL_ARROW	Control block eyecatcher
(3)	CHARACTER	3	GLOBAL_DFH	
(6)	CHARACTER	2	GLOBAL_DOMAIN	
(8)	CHARACTER	8	GLOBAL_BLOCK_ID	

Loader Domain Status Flags

Offset Hex	Type	Len	Name (Dim)	Description
(10)	BITSTRING	2	LD_DOMAIN_STATUS	Status of Loader domain
(12)	UNSIGNED	1	LD_RPL_STATUS	Status of DFHRPL library
(13)	UNSIGNED	1	LD_LPA_STATUS	Status of LPA
(14)	BITSTRING	1	LD_FLAGS	Loader global flags
	1...		LD_GLOBAL_ CATALOG_IN_USE	GCD in use
	.1..		LD_CICS_ INITIALISED	CICS fully up
	..1.		LD_CICS_ COLD_STARTED	CICS cold started
	...1		LD_LLACOPY_ IN_REFRESH	
	.... 1...		LD_XLDLOAD_ ACTIVE	
	.... .1..		LD_XLDELETE_ ACTIVE	
	11		*	Reserved
(15)	BITSTRING	1	*	Reserved
(16)	UNSIGNED	1	LD_LLACOPY_ STATUS	LLACOPY status
(17)	UNSIGNED	1	LD_SLD	SLD support?

Storage Manager subpool tokens for Loader managed subpools.

Offset Hex	Type	Len	Name (Dim)	Description
(18)	CHARACTER	8	LD_CONTROL_POOL	Control subpool token

Offset Hex	Type	Len	Name (Dim)	Description
(20)	CHARACTER	8	LD_APE_CELL_POOL	APE subpool token
(28)	CHARACTER	8	LD_CSECTL_ CELL_POOL	CSECTL subpool token
(30)	CHARACTER	8	LD_CPE_CELL_POOL	CPE subpool token
(38)	CHARACTER	8	LD_DUMMY_ CDE_POOL	DUMMY_CDE subpool token
(40)	CHARACTER	12	LD_SUBPOOL_DATA2 (12)	Array of program subpools
(40)	CHARACTER	8	TOKEN2	Subpool token
(48)	UNSIGNED	4	DSA2	DSA identifier

Lock tokens

Offset Hex	Type	Len	Name (Dim)	Description
(D0)	ADDRESS	4	LD_STATE_LOCK	Loader state lock token
(D4)	ADDRESS	4	LD_LIBRARY_LOCK	Loader libraray lock token

Loader chains
There are six Loader chains anchored in Global storage:

1. Global CPE chain - this contains all the CPEs (in alphabetical order of program name) for all the programs currently defined to the system.
2. Global APE chain - this contains an APE for every program instance currently residing in CICS storage. Entries are in ascending order of entry point.
3. The APE NIU chain - contains all the APEs associated with programs defined as REUSABLE which have a use count of zero. These programs are eligible to be removed on program compression.
4. The LDWE chain - this contains a Loader Wait Element for each task that has been suspended due to a CPE being locked. LDWEs are added to the top of the chain.
5. The LDBE chain - this chain contains a Loader Browse Element for each currently active browse session. New entries are added to the head of the chain.
6. The DUMMY_CDE chain - managed on behalf of SLD. There is one CDE per loaded program.

Offset Hex	Type	Len	Name (Dim)	Description
(D8)	FULLWORD	4	CPE_CHAIN_SIZE	Global CPE chain size
(DC)	CHARACTER	24	CPE_ANCHOR	
(F4)	FULLWORD	4	APE_CHAIN_SIZE	Global APE chain size
(F8)	FULLWORD	4	APE_NIU_ CHAIN_SIZE	APE NIU chain size
(FC)	CHARACTER	48	APE_ANCHOR	
(12C)	FULLWORD	4	LDWE_CHAIN_SIZE	LDWE chain size
(130)	CHARACTER	24	LDWE_ANCHOR	
(148)	FULLWORD	4	LDBE_CHAIN_SIZE	LDBE chain size
(14C)	CHARACTER	24	LDBE_ANCHOR	
(164)	CHARACTER	24	DUMMY_CDE_ANCHOR	Dummy CDE chain
(17C)	ADDRESS	4	PRVMOD_PTR	-> area holding prog names that require loading from RPL rather than LPA

Global statistics

Offset Hex	Type	Len	Name (Dim)	Description
(180)	ADDRESS	4	LD_STATS_ BUFFER_PTR	-> Loader stats buffer
(184)	FULLWORD	4	STA_DEFINES	No. of DEFINE_PROGRAMs
(188)	FULLWORD	4	STA_DELETES	No. of DELETE_PROGRAMs
(18C)	FULLWORD	4	STA_INQUIRES	No. of INQUIRE_PROGRAMs
(190)	FULLWORD	4	STA_REFRESHS	No. of REFRESH_PROGRAMs
(194)	FULLWORD	4	STA_BROWSES	No. of START_BROWSEs
(198)	FULLWORD	4	STA_NOTIFIES	No. of SM notify calls received.

long name cache stats
No. of times long name longer than cache key length

Offset Hex	Type	Len	Name (Dim)	Description
(19C)	FULLWORD	4	STA_NAME2LONG	

Length of longest name given to CONVERT_ NAME

Offset Hex	Type	Len	Name (Dim)	Description
(1A0)	FULLWORD	4	STA_LONGEST_NAME	

No. of adds to cache = max cache size

Offset Hex	Type	Len	Name (Dim)	Description
(1A4)	FULLWORD	4	STA_NAME_ADDED	
(1A8)	FULLWORD	4	*	Reserved
(1AC)	FULLWORD	4	STA_FETCHS	No. of loads from the RPL library
(1B0)	FULLWORD	4	STA_FETCH_TIME	Total fetch time
(1B4)	FULLWORD	4	STA_USES	Total no. of times progs are reused
(1B8)	FULLWORD	4	STA_WAITS	No. of tasks currently suspended
(1BC)	FULLWORD	4	STA_WAITS_PAST	Total no. of tasks suspended
(1C0)	FULLWORD	4	STA_WAITS_HWM	High Water Mark for STA_WAITS.
(1C4)	FULLWORD	4	STA_TIMES_ WAITS_HWM	No. of times High Water Mark is reached
(1C8)	FULLWORD	4	STA_WAIT_TIME	Total time tasks are suspended.
(1CC)	FULLWORD	4	STA_DEB_REBUILDS	No. of times DEB rebuilt due to an extent error
(1D0)	CHARACTER	8	STA_LAST_ RESET_TIME	Time stats last reset
(1D8)	FULLWORD	4	LD_STORAGE_ FACTOR	Loader storage factor
(1DC)	CHARACTER	32	LD_DSA_RECORDS (6)	Array showing storage usage for each DSA
(1DC)	FULLWORD	4	LD_DSA_USAGE	Storage used
(1E0)	FULLWORD	4	LD_DSA_RPS	Redundant program storage
(1E4)	FULLWORD	4	LD_DSA_TARGET	Target storage level
(1E8)	FULLWORD	4	LD_DSA_ PROG_REMOVALS	Number of programs removed by DPSC
(1EC)	FULLWORD	4	LD_DSA_RECLAIMS	Number of programs reclaimed from RPS

Offset Hex	Type	Len	Name (Dim)	Description
(1F0)	CHARACTER	8	LD_DSA_ NIU_Q_TIME	Total time spent on NIU queue
(1F8)	FULLWORD	4	LD_DSA_ NIU_Q_SIZE	NIU queue size
Loader generic gate entry points				
(29C)	ADDRESS	4	LD_NT_EPADDR	SMNT gate
(2A0)	ADDRESS	4	LD_ST_EPADDR	STST gate
(2A4)	ADDRESS	4	LD_DC_EPADDR	Dynamic compression routine
DFHSIP entry point address				
(2A8)	ADDRESS	4	LD_DFHSIP_EPADDR	DFHSIP entry point
(2AC)	FULLWORD	4	*	reserved
(2B0)	FULLWORD	4	*	reserved
(2B4)	FULLWORD	4	*	reserved
(2B8)	FULLWORD	4	*	reserved
(2BC)	CHARACTER	12	LD_SUBPOOL_DATA (16)	Array of program subpools
(2BC)	CHARACTER	8	TOKEN	Subpool token
(2C4)	UNSIGNED	4	DSA	DSA identifier
(37C)	FULLWORD	4	*	reserved
(380)	FULLWORD	4	*	reserved
(384)	FULLWORD	4	*	reserved
(388)	FULLWORD	4	*	reserved
Long Name cache directory token				
(38C)	ADDRESS	4	LD_LONG_NAME_CACHE_TOKEN	
(390)	CHARACTER		*	

LAFPB - LOADER AUTHORISED FACILITIES PARAMETER BLOCK

The LAFPB contains the authorised function code, the return code, the BLDL parameter list used by LLACOPY, the program length (LPA load only), the entry point of the module (LPA load only) and the creation time of the LAFPB.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	LAFPB	
(0)	CHARACTER	16	LAFPB_PREFIX	Control block prefix
(0)	UNSIGNED	2	LAFPB_LENGTH	Control block length
(2)	CHARACTER	1	LAFPB_ARROW	Control block eyecatcher
(3)	CHARACTER	3	LAFPB_DFH	
(6)	CHARACTER	2	LAFPB_DOMAIN	
(8)	CHARACTER	8	LAFPB_BLOCK_ID	
(10)	UNSIGNED	1	LAFPB_FUNCTION	Required auth. function
(11)	UNSIGNED	1	LAFPB_RESPONSE	Response from function
(12)	UNSIGNED	2	*	Reserved
Abend data saved on a LOAD failure				
(14)	UNSIGNED	2	LAFPB_ABEND	
(16)	UNSIGNED	2	LAFPB_REASON	
(18)	UNSIGNED	4	LAFPB_R0	
The following fields are used for RPL loads. For DISCONNECT, LAFPB_ BLDL_PLIST contains the MLTK. For GET_ SMDE, LAFPB_ BLDL_PLIST points at name list. For LOAD_ WITH PMARL, the PMARL is returned in LAFPB_ DESERV_AREA For END, LAFPB_ DESERV_ AREA addresses the Loader Information Table, mapped by IEWBLIT.				
(1C)	ADDRESS	4	LAFPB_BLDL_PLIST	-> to BLDL_LIST
(20)	ADDRESS	4	LAFPB_LOAD_POINT	-> for directed load
(24)	CHARACTER	8	LAFPB_CREATION_ STCK	time LAFPB created
(2C)	ADDRESS	4	LAFPB_DESERV_ AREA	-> space for result
(30)	FULLWORD	4	LAFPB_DESERV_ AREAL	length of result area
(34)	CHARACTER		*	

LDBE - LOADER DOMAIN BROWSE ELEMENT

The LDBE represents a browse session. It contains the adress of the last CPE browsed, the program name from the last CPE browsed, the address of the last APE browsed, the entry point address from the last APE browsed and the creation time of the LDBE.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	LDBE	
(0)	CHARACTER	24	LDBE_PREFIX	Control block prefix
(0)	UNSIGNED	2	LDBE_LENGTH	Control block length
(2)	CHARACTER	1	LDBE_ARROW	Control block eyecatcher
(3)	CHARACTER	3	LDBE_DFH	
(6)	CHARACTER	2	LDBE_DOMAIN	
(8)	CHARACTER	8	LDBE_BLOCK_ID	
(10)	ADDRESS	4	LDBE_NEXT	-> next LDBE in chain
(14)	ADDRESS	4	LDBE_PRIOR	-> previous LDBE in chain
(18)	ADDRESS	4	LDBE_LAST_CPE_ADDRESS	Addr last CPE browsed
(1C)	ADDRESS	4	LDBE_LAST_APE_ADDRESS	Addr last APE browsed
(20)	ADDRESS	4	LDBE_LAST_ENTRY_POINT	Entry point from APE
(24)	CHARACTER	8	LDBE_LAST_PROGRAM_NAME	Program name from CPE
(2C)	CHARACTER	8	LDBE_CREATION_STCK	Time LDBE was created
(34)	CHARACTER		*	

LDWE - LOADER DOMAIN WAIT ELEMENT

The LDWE represents a task that has been suspended because the CPE it requires is currently locked. The LDWE contains the name of the program the task is waiting for, the associated suspend token and the time the LDWE was created.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	LDWE	
(0)	CHARACTER	24	LDWE_PREFIX	Control block prefix
(0)	UNSIGNED	2	LDWE_LENGTH	Control block length
(2)	CHARACTER	1	LDWE_ARROW	Control block eyecatcher
(3)	CHARACTER	3	LDWE_DFH	
(6)	CHARACTER	2	LDWE_DOMAIN	
(8)	CHARACTER	8	LDWE_BLOCK_ID	
(10)	ADDRESS	4	LDWE_NEXT	-> next LDWE on chain
(14)	ADDRESS	4	LDWE_PRIOR	-> previous LDWE on chain
(18)	ADDRESS	4	LDWE_SUSPEND_TOKEN	Dispatcher suspend token
(1C)	ADDRESS	4	LDWE_CPE_ADDRESS	Addr. of locked CPE
(20)	CHARACTER	8	LDWE_PROGRAM_NAME	Name of program
(28)	CHARACTER	8	LDWE_CREATION_STCK	Time LDWE created
(30)	FULLWORD	4	LDWE_RESUME_REQUIRED	Flag to indicate task requires resuming
(34)	CHARACTER		*	

LOB - LOADER OPTION BLOCK

The LOB is used to save Loader SIT parameters (LPA usage and storage factor) and the sizes of the resident subpools. These figures are used on restart. It should be noted that irregardless of the type of start Loader always attempts to read the LOB from the catalog.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	LOB	
(0)	FULLWORD	4	LOB_STORAGE_FACTOR	Loader storage factor
(4)	UNSIGNED	1	LOB_LPA_STATUS	LPA status
(5)	UNSIGNED	1	LOB_LLACOPY_STATUS	
(6)	CHARACTER	2	*	

The resident subpool sizes. These are read from the catalog at initialisation and used to recreate the subpools with the same INITIAL_FREE size as on the previous CICS run.

Offset Hex	Type	Len	Name (Dim)	Description
(8)	UNSIGNED	4	LOB_APE_CELL_POOL_SIZE	APE subpool size
(C)	UNSIGNED	4	LOB_CSECTL_CELL_POOL_SIZE	CSECTL subpool size
(10)	CHARACTER	8	LOB_CREATION_STCK	Time LOB created

Offset Hex	Type	Len	Name (Dim)	Description
(18)	CHARACTER		*	

P D B - PROGRAM DESCRIPTOR BLOCK

A PDB describes a programs attributes.It is this control block that is written to one of the catalogs each time a program is defined (unless CATALOG_ MODULE(NO) is specified). On restart the PDBs are retrieved from the catalogs and CPEs are built.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	PDB	
(0)	CHARACTER	16	PDB_DESCRIPTOR_ FIELDS	

PROGRAM_ TYPE maybe PRIVATE, SHARED or TYPE_ ANY. PRIVATE means the program will always be loaded into CICS managed storage. SHARED means the program resides in the LPA. TYPE_ ANY means an LPA version of the program will be used if there is one otherwise an RPL version will be loaded.

(0)	UNSIGNED	1	PDB_PROGRAM_ TYPE	Where to load the program from

PROGRAM_USAGE maybe NUCLEUS or APPLICATION. If NUCLEUS is specified the PDB will be written to the LCD. If APPLICATION is specified the PDB will be written to the GCD.

(1)	UNSIGNED	1	PDB_PROGRAM_ USAGE	Where to catalog the definition

PROGRAM_ATTRIBUTE maybe RESIDENT, REUSABLE, TRANSIENT or RELOAD RESIDENT programs must be at least quasi- reentrant and are not eligible program compression.REUSABLE programs must be at least quasi_reentrant and are eligible for program compression. TRANSIENT programs must be at least quasi_reentrant and are removed from storage as soon as the use count reaches zero. RELOAD programs do not need to be reentrant a new version of the program is loaded each time the program is ACQUIREd. Such a program is removed from storage when it is RELEASEd.

(2)	UNSIGNED	1	PDB_PROGRAM_ ATTRIBUTE	Prog load attribute
(3)	UNSIGNED	1	PDB_REQUIRED_ RMODE	RMODE of the program, 24, ANY or default
(4)	UNSIGNED	1	PDB_REQUIRED_ AMODE	AMODE of the program 31 24,ANY or default@P3A
(5)	UNSIGNED	1	PDB_CATALOG_ MODULE	Indicates whether PDB should be cataloged
(6)	UNSIGNED	1	PDB_EXECUTION_ KEY	EXECKEY of the program, CICS or USER
(7)	CHARACTER	1	*	reserved
(8)	CHARACTER	8	PDB_CREATION_ STCK	Time PDB created
(10)	CHARACTER		*	

Constants

Len	Type	Value	Name	Description
8	CHARACTER	APE	APE_ID_STRING	
8	CHARACTER	APE-ANCH	APE_ANCHOR_ID	
8	CHARACTER	CPE-APE	CPE_APE_ANCHOR_ID	

APE status

1	HEX	80	APE_ACTIVE	
1	HEX	FF	APE_FREED	

BLDL associated constants.

8	CHARACTER	BLDL_LST	BLDL_ID_STRING	

CPE associated constants.

8	CHARACTER	CPE	CPE_ID_STRING	
8	CHARACTER	CPE-ANCH	CPE_ANCHOR_ID	

CPE program status

1	HEX	00	CPE_UNUSED	Program defined
1	HEX	01	CPE_LOCATED	Program defined and located
1	HEX	02	CPE_LOADED	Program defined, located and loaded
1	HEX	0F	CPE_DELETED	Program definition deleted
1	HEX	F0	CPE_BAD	Corrupt CPE
1	HEX	FF	CPE_FREED	CPE freemained

CPE catalog status

1	HEX	03	CPE_CC_DONE	PDB cataloged

Len	Type	Value	Name	Description
1	HEX	04	CPE_CC_REQD	PDB requires cataloging
CPE lock values. APE_CREATING and CSECTL_CREATING can occur while other CPE locks are held. They are added to the existing locks temporarily when SOS and a GETMAIN with SUSPEND(YES) is issued. Therefore, X'10'to X'13' and X'20' to X'23' are reserved.				
1	HEX	00	CPE_UNLOCKED	CPE not being updated.
1	HEX	01	CPE_LPA_LOCATING	Program being located in LPA
1	HEX	02	CPE_RPL_LOCATING	Program being located in RPL
1	HEX	03	CPE_RPL_LOADING	Program being loaded from RPL
1	HEX	04	CPE_DISCONNECTING	RPL member being disconnected
1	HEX	10	CPE_APE_CREATING	APE being created for CPE
1	HEX	20	CPE_CSECTL_ CREATING	CSECT lists being created P7A
6	CHARACTER	LDCPE	CPE_EYE_CATCH_I	
CSECTL associated constants				
8	CHARACTER	CSECTL	CSECTL_ID_STRING	
1	DECIMAL	4	CSECTL_NUMBER_OF_ENTRIES	
8	CHARACTER	DUMMYCDE	CDE_ID_STRING	
D U M P - DUMP CONTROL RECORD IDENTIFIERS ------- These are the dump record identifiers and names for items dumped by Loader dump subroutine.				
8	CHARACTER	LD0001	LDDU_ABEND	
Abend detected in module				
8	CHARACTER	LD0002	LDDU_SEVERE_ERROR	
Severe error detected				
8	CHARACTER	LD0004	LDDU_LOOP	
Loop detected in module				
8	CHARACTER	LD0105	LDDU_BAD_LOB	
Corrupt LOB detected				
8	CHARACTER	LD0201	LDDU_BAD_STRUCTURE	
Corrupt CPE detected				
8	CHARACTER	LD0204	LDDU_BAD_PDB	
Global associated constants				
8	CHARACTER	ANCHOR	GLOBAL_ID_STRING	
2	CHARACTER	LD	EYECATCHER_DOMID	
3	CHARACTER	DFH	EYECATCHER_DFH	
1	CHARACTER	>	EYECATCHER_ARROW	
Program subpool constants				
4	DECIMAL	16	MAXSUBPOOLS	
4	DECIMAL	1	NUCLEUS24_POOL	
4	DECIMAL	2	NUCLEUS31_POOL	
4	DECIMAL	3	NUCLEUS24_RO_POOL	
4	DECIMAL	4	NUCLEUS31_RO_POOL	
4	DECIMAL	5	NUCLEUS24_RESIDENT_POOL	
4	DECIMAL	6	NUCLEUS31_RESIDENT_POOL	
4	DECIMAL	7	NUCLEUS24_RESIDENT_RO_POOL	
4	DECIMAL	8	NUCLEUS31_RESIDENT_RO_POOL	
4	DECIMAL	9	RESIDENT24_POOL	
4	DECIMAL	10	RESIDENT31_POOL	
4	DECIMAL	11	RESIDENT24_RO_POOL	
4	DECIMAL	12	RESIDENT31_RO_POOL	
4	DECIMAL	13	PROGRAM24_POOL	
4	DECIMAL	14	PROGRAM31_POOL	
4	DECIMAL	15	PROGRAM24_RO_POOL	
4	DECIMAL	16	PROGRAM31_RO_POOL	
Storage subpool ID strings				
8	CHARACTER	LD_CNTRL	CONTROL_POOL_NAME	
8	CHARACTER	LD_APES	APE_CELL_POOL_NAME	
8	CHARACTER	LD_CPES	CPE_CELL_POOL_NAME	
8	CHARACTER	LD_CSECT	CSECTL_CELL_ POOL_NAME	
8	CHARACTER	LD_CDE	DUMMY_CDE_ POOL_NAME	
8	CHARACTER	LDNUC	NUCLEUS24_ POOL_NAME	
8	CHARACTER	LDENUC	NUCLEUS31_ POOL_NAME	
8	CHARACTER	LDNUCRO	NUCLEUS24_RO_POOL_NAME	
8	CHARACTER	LDENUCRO	NUCLEUS31_RO_POOL_NAME	
8	CHARACTER	LDNRS	NUCLEUS24_RESIDENT_POOL_ NAME	

Len	Type	Value	Name	Description
8	CHARACTER	LDENRS	NUCLEUS31_RESIDENT_POOL_ NAME	
8	CHARACTER	LDNRSRO	NUCLEUS24_RESIDENT_RO_POOL_NAME	
8	CHARACTER	LDENRSRO	NUCLEUS31_RESIDENT_RO_POOL_NAME	
8	CHARACTER	LDRES	RESIDENT24_ POOL_NAME	
8	CHARACTER	LDERES	RESIDENT31_ POOL_NAME	
8	CHARACTER	LDRESRO	RESIDENT24_RO_POOL_NAME	
8	CHARACTER	LDERESRO	RESIDENT31_RO_POOL_NAME	
8	CHARACTER	LDPGM	PROGRAM24_ POOL_NAME	
8	CHARACTER	LDEPGM	PROGRAM31_ POOL_NAME	
8	CHARACTER	LDPGMRO	PROGRAM24_RO_POOL_NAME	
8	CHARACTER	LDEPGMRO	PROGRAM31_RO_POOL_NAME	
Storage subpool boundary constants				
2	DECIMAL	16	CONTROL_POOL_BDY	
2	DECIMAL	8	APE_CELL_POOL_BDY	
2	DECIMAL	8	CPE_CELL_POOL_BDY	
2	DECIMAL	8	CSECTL_CELL_ POOL_BDY	
2	DECIMAL	16	DUMMY_CDE_POOL_BDY	
2	DECIMAL	16	NUCLEUS_POOLS_BDY	
2	DECIMAL	16	RESIDENT_POOLS_BDY	
2	DECIMAL	16	PROGRAM_POOLS_BDY	
Number of DSAs. Note that Loader does not load programs into all DSAs.				
4	DECIMAL	6	MAXDSAS	
5	CHARACTER	CDSA	CDSA_NAME	
5	CHARACTER	SDSA	SDSA_NAME	
5	CHARACTER	RDSA	RDSA_NAME	
5	CHARACTER	ECDSA	ECDSA_NAME	
5	CHARACTER	ESDSA	ESDSA_NAME	
5	CHARACTER	ERDSA	ERDSA_NAME	
5	CHARACTER	LPA	LPA_NAME	
5	CHARACTER	ELPA	ELPA_NAME	
5	CHARACTER	RGN	RGN_NAME	
5	CHARACTER	ERGN	ERGN_NAME	
Loader domain statuses				
2	DECIMAL	1023	LOADER_PRE_ INITIALISING	
2	DECIMAL	1024	LOADER_PRE_ INITIALISED	
2	DECIMAL	2047	LOADER_INITIALISING	
2	DECIMAL	2048	LOADER_UP_AND_RUNNING	
2	DECIMAL	28671	LOADER_QUIESCING	
2	DECIMAL	28672	LOADER_QUIESCED	
2	DECIMAL	32767	LOADER_TERMINATING	
2	DECIMAL	32768	LOADER_TERMINATED	
LPA statuses				
1	DECIMAL	2	LD_LPA_NOT_IN_USE	
1	DECIMAL	1	LD_LPA_IN_USE	
DFHRPL library statuses				
1	HEX	FF	LD_RPL_CLOSED	
1	HEX	A1	LD_RPL_OPEN	
LLACOPY usage status				
1	DECIMAL	1	LD_LLACOPY_YES	
1	DECIMAL	2	LD_LLACOPY_NO	
1	DECIMAL	3	LD_LLACOPY_NEWCOPY	
Loader domain lock data				
8	CHARACTER	LD_GBLOK	STATE_LOCK_NAME	
8	CHARACTER	LD_LBLOK	LIBRARY_LOCK_NAME	
Loader CICS catalog record types				
8	CHARACTER	LD_PDEFN	PROGRAM_DEFINITION	
8	CHARACTER	LD_LOB	OPTION_BLOCK	
Loader loaded modules				
8	CHARACTER	DFHLDDMI	SECONDARY_INITIALISATION	
8	CHARACTER	DFHLDNT	STORAGE_NOTIFY	
8	CHARACTER	DFHLDST	STATISTICS	
Default definitions				
1	DECIMAL	3	DEFAULT_PROGRAM_ TYPE	

Len	**Type**	**Value**	**Name**	**Description**
1	DECIMAL	1	DEFAULT_PROGRAM_ USAGE	
1	DECIMAL	2	DEFAULT_PROGRAM_ ATTRIBUTE	
1	DECIMAL	3	DEFAULT_REQUIRED_ RMODE	
1	DECIMAL	4	DEFAULT_REQUIRED_ AMODE	
1	DECIMAL	1	DEFAULT_CATALOG_ MODULE	
1	DECIMAL	2	DEFAULT_EXECUTION_ KEY	
4	DECIMAL	16777216	DEFAULT_DSA_ RPS_TARGET	
4	DECIMAL	2147483647	DEFAULT_EDSA_ RPS_TARGET	
1	DECIMAL	50	DEFAULT_STORAGE_ FACTOR	
Miscellaneous constants				
4	HEX	00FFFFFF	SIXTEEN_MEG	
4	DECIMAL	14336	LD_STATS_ BUFFER_SIZE	
4	CHARACTER	LDNM	LD_LONG_NAME_ CACHE_NAME	
4	DECIMAL	64	LD_LONG_NAME_ CACHE_KEYL	
Cache entry data (ETOKEN) contains a member name, or the following value to show that DESERV couldn't find the alias.				
8	CHARACTER		LD_LONG_NAME_ NOT_IN_RPL	
or the following value to show that the cache has been told to forget, during a NEWCOPY.				
8	CHARACTER		LD_LONG_NAME_ CACHE_INVALID	
The following value is used in the code to remember that there was no entry in the cache for a given name.				
8	CHARACTER		LD_LONG_NAME_ NOT_CACHED	
LPA search routine responses				
1	DECIMAL	8	NOT_FOUND	
LAFPB associated constants				
8	CHARACTER	LAFPB	LAFPB_ID_STRING	
LAFPB function codes				
1	DECIMAL	1	LAFPB_RPL_LOAD	
1	DECIMAL	2	LAFPB_RPL_BLDL	
1	DECIMAL	4	LAFPB_RPL_OPEN	
1	DECIMAL	8	LAFPB_RPL_CLOSE	
1	DECIMAL	16	LAFPB_RPL_LLACOPY	
1	DECIMAL	32	LAFPB_RPL_ DISCONNECT	
1	DECIMAL	33	LAFPB_RPL_GET_SMDE	
1	DECIMAL	34	LAFPB_RPL_ LOAD_WITH_PMAR	
1	DECIMAL	35	LAFPB_RPL_END	
LAFPB response codes				
1	DECIMAL	0	LAFPB_OK	
1	DECIMAL	1	LAFPB_NOTFOUND	
1	DECIMAL	2	LAFPB_NOT_ EXECUTABLE	
1	DECIMAL	3	LAFPB_IOERR	
1	DECIMAL	4	LAFPB_NOSTORE	
1	DECIMAL	5	LAFPB_OPEN_ERROR	
1	DECIMAL	6	LAFPB_CLOSE_ERROR	
1	DECIMAL	8	LAFPB_EXTENT_ERROR	
1	DECIMAL	9	LAFPB_NOT_ CONNECTED	
1	DECIMAL	10	LAFPB_IS_PDS	
1	DECIMAL	11	LAFPB_BAD_CONCATNO	
1	DECIMAL	12	LAFPB_INFO	
1	DECIMAL	13	LAFPB_WARN	
1	DECIMAL	14	LAFPB_PARM	
1	DECIMAL	15	LAFPB_CALR	
1	DECIMAL	16	LAFPB_NO_FESTAE	
1	DECIMAL	17	LAFPB_ENVR	
1	DECIMAL	18	LAFPB_BAD_PARM	
1	DECIMAL	32	LAFPB_NO_DD	
1	DECIMAL	64	LAFPB_NO_ AUTHORISATION	
1	DECIMAL	65	LAFPB_BAD_STORAGE	
1	DECIMAL	128	LAFPB_UNKNOWN_ ERROR	
1	DECIMAL	255	LAFPB_INVALID_ FUNCTION	
LDBE associated constants				
8	CHARACTER	LDBE	LDBE_ID_STRING	

Len	Type	Value	Name	Description
8	CHARACTER	LDBE_ANC	LDBE_ANCHOR_ID	
	LDWE associated constants			
8	CHARACTER	LDWE	LDWE_ID_STRING	
8	CHARACTER	LDWE_ANC	LDWE_ANCHOR_ID	
4	DECIMAL	0	LDWE_RESUME_NO	Resume not required
4	DECIMAL	1	LDWE_RESUME_YES	Resume required
	Abend detected in LD module Insert1_hex = offset from module start Insert2_char = Module name			
4	DECIMAL	1	LDME_ABEND	
	Severe error detected in LD module Insert1_ hex = offset from module start Insert2_ char = Module name			
4	DECIMAL	2	LDME_SEVERE_ERROR	
	Loop detected in LD module Insert1_ hex = offset from module start Insert2_ char = Module name			
4	DECIMAL	4	LDME_LOOP	
	Loader nucleus module not found. Insert1_ char = Module name			
4	DECIMAL	101	LDME_NO_MODULE	
	Unable to define entry point to DFHLDNT module. Insert1_ char = Module name Insert2_ char = Format number			
4	DECIMAL	102	LDME_ADD_ GATE_FAILED	
	Dynamic program storage compression is not operational, all non-resident programs will be treated as USAGE=TRANSIENT.			
4	DECIMAL	103	LDME_NO_NT_MODULE	
	Program statistics are not being collected.			
4	DECIMAL	104	LDME_NO_ST_MODULE	
	The Loader Option Block (LOB) read from the CICS catalog contains at least one invalid field. All parameters in this block have been ignored.			
4	DECIMAL	105	LDME_CC_LOB_BAD	
	Bad response 'hh'x received when attempting to open the relocatable library (DFHRPL). Insert1_ bin = I/O error response from DCB			
4	DECIMAL	106	LDME_BAD_OPEN	
	The Link Pack Area (LPA) has been searched for a given module, no module was found. Loader domain will now search the Relocatable Program Library (RPL).			
4	DECIMAL	107	LDME_NOT_IN_LPA	*
	@BA57063A The maximum number of entries, 32767, to @BA57063A be passed to BLDL on the BLDL parameter @BA57063A list, has been exceeded. @BA57063A			
4	DECIMAL	108	LDME_BLDL_ LIMIT_EXCEEDED	@BA57063A
	Invalid PROGRAM_TYPE field detected in Loader 'BBB' structure at location 'hhhhhhhh'. ---diagnosis-- ---diagnosis--- is one of the following texts: 1/ (Storage overwrite suspected.) 2/ (Catalog corruption suspected.) Insert1_char = Blockid (PDB,CPE or APE) Insert2_bin = address of control block in error.			
4	DECIMAL	201	LDME_CONBLOK_ INVALID	
	SVC request failed due to shortage of OS storage.			
4	DECIMAL	202	LDME_NO_OS_STORAGE	
	SVC request failed due to library I/O errors.			
4	DECIMAL	203	LDME_LIBRARY_ IO_ERROR	
	Bad Loader PDB for program 'progname' read from Global|Local catalog, corruption suspected. Insert1_ char = program name Insert2_ bin = optional text number			
4	DECIMAL	204	LDME_BAD_PDB	*
1	DECIMAL	2	ME_GLOBAL_CAT	
1	DECIMAL	1	ME_LOCAL_CAT	
	DOMAIN ENTRY (LDLD functional gate) level = 1 module = DFHLDLD Generated as the first operation on entry to the domain for all calls. caller. DATA1 = Loader Parameter list			
2	HEX	0001	TRLD_ENTRY_TRACE	
	DOMAIN EXIT (LDLD functional gate) level = 1 or EXCEPTION module = DFHLDLD Generated as the final operation prior to performing return via the Kernel to the Loader's caller. DATA1 = Loader Parameter list			
2	HEX	0002	TRLD_EXIT_TRACE	

Len	Type	Value	Name	Description
RECOVERY ENTERED (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data				
2	HEX	0701	TRLD_RECOVERY_ENTERED	
INVALID FORMAT (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a call is made to the LDLD gate using the incorrect parameter list format DATA1 = parameter list				
2	HEX	0801	TRLD_INVALID_ FORMAT	
INVALID FUNCTION (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a call is made to the LDLD gate specifying an invalid function. DATA1 = parameter list				
2	HEX	0802	TRLD_INVALID_ FUNCTION	
INVALID PARAMETERS (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if an invalid combination of parameters is detected. DATA1 = parameter list				
2	HEX	0803	TRLD_INVALID_PARAMETERS	
INVALID PDB (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if an invalid PDB is recovered from the catalog. DATA1 = program name DATA2 = PDB				
2	HEX	0804	TRLD_BAD_PDB	
INVALID ENTRY POINT (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if an invalid entry point is presented to the Loader on a release program request. DATA1 = call plist				
2	HEX	0806	TRLD_INVALID_ENTRY_POINT	
INVALID PGM TOKEN (LDLD FUNCTIONAL GATE) level = EXCEPTION module = DFHLDLD These trace entries are put out if an invalid program token is presented to the loader. DATA2 = CALL PLIST				
2	HEX	0807	TRLD_INVALID_PGM_TOKEN	
2	HEX	0808	TRLD_INVALID_PGM_TOKEN_1	
2	HEX	0809	TRLD_INVALID_PGM_TOKEN_2	
LDWE GET FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a GETMAIN for an LDWE fails whilst trying to suspend a task. DATA1 = parameter list				
2	HEX	0903	TRLD_LDWE_GETMAIN	
ADD SUSPEND FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a dispatcher ADD_ SUSPEND request fails whilst trying to suspend a task due to a CPE having been locked by another task in the system for a LOAD or BLDL. DATA1 = parameter list				
2	HEX	0905	TRLD_ADD_SUSPEND	
DELETE SUSPEND FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a dispatcher DELETE_ SUSPEND request fails whilst trying to suspend a task due to a CPE having been locked by another task in the system for a LOAD or BLDL. DATA1 = parameter list				
2	HEX	0906	TRLD_DELETE_ SUSPEND	
SUSPEND FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a dispatcher SUSPEND request fails whilst trying to suspend a task due to a CPE having been locked by another task in the system for a LOAD or BLDL. DATA1 = parameter list				
2	HEX	0907	TRLD_SUSPEND	
CPE GETMAIN (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a getmain for storage for a CPE fails. DATA1 = parameter list				
2	HEX	0908	TRLD_CPE_GETMAIN	
LOCK FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD These trace entries are put out if a request to LOCK the Loader state lock fails. DATA1 = parameter list				
2	HEX	0909	TRLD_LOCK	
2	HEX	090A	TRLD_LOCK_1	

Len	Type	Value	Name	Description
UNLOCK FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD These trace entries are put out if a request to UNLOCK the Loader state lock fails. DATA1 = parameter list				
2	HEX	090B	TRLD_UNLOCK	
2	HEX	090C	TRLD_UNLOCK_1	
INQUIRE START (LDLD functional gate) level = EXCEPTION module = DFHLDLD This trace entry is put out if a request to Parameter Domain to determine CICS Start type fails. DATA1 = PAGP parameter list DATA2 = LDLD parameter list				
2	HEX	090D	TRLD_INQUIRE_START	
PRE-SVC CALL (LDLD functional gate) level = 1 module = DFHLDLD1 Generated immediately prior to issuing an SVC to invoke the Loader's authorised facilities module. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load)				
2	HEX	1003	TRLD1_SVC_CALL	
PRE-SVC CALL (LDLD functional gate) level = 1 module = DFHLDLD2 Generated immediately prior to issuing an SVC to invoke the Loader's authorised facilities module. DATA1 = Authorised Facility Plist				
2	HEX	2904	TRLD2_SVC_CALL	
PRE-SVC CALL (LDLD functional gate) level = 1 module = DFHLDLD3 Generated immediately prior to issuing an SVC to invoke the Loader's authorised facilities module. DATA1 = Authorised Facility Plist < DATA2 > = DESERV GET Name List (DESN)				
2	HEX	390B	TRLD3_SVC_CALL	
POST-SVC CALL (LDLD functional gate) level = 1 module = DFHLDLD1 Generated on return from the Loader's SVC service routine if a normal return code has been presented by the routine. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load)				
2	HEX	1004	TRLD1_SVC_RETURN	
POST-SVC CALL (LDLD functional gate) level = 1 module = DFHLDLD2 Generated on return from the Loader's SVC service routine if a normal return code has been presented by the routine. DATA1 = Authorised Facility Plist				
2	HEX	2905	TRLD2_SVC_RETURN	
POST-SVC CALL (LDLD functional gate) level = 1 module = DFHLDLD3 Generated on return from the Loader's SVC service routine if a normal return code has been presented by the routine. DATA1 = Authorised Facility Plist < DATA2 > = DESERV GET o/p area (DESB)				
2	HEX	390C	TRLD3_SVC_RETURN	
DSA_ COMPRESSION (LDLD functional gate) level = 2 module = DFHLDLD1 Generated when a program instance is selected for deletion from a DSA by the program storage compression algorithms. DATA1 = Active Program Element (APE) DATA2 = DSA name				
2	HEX	1005	TRLD1_DSA_ COMPRESSION	
PRE_ LOAD (LDLD functional gate) level = 1 module = DFHLDLD1 Generated prior to issuing a CSVQUERY call to access an LPA resident module. DATA1 = Program name				
2	HEX	1007	TRLD1_PRE_CSVQUERY	
POST_ LOAD (LDLD functional gate) level = 1 module = DFHLDLD1 Generated after issuing a CSVQUERY call to access an LPA resident module. DATA1 = Program name DATA1 = Return code				
2	HEX	1008	TRLD1_POST_ CSVQUERY	
RECOVERY_ ENTERED (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data				
2	HEX	1701	TRLD1_RECOVERY_ ENTERED	
INVALID_ FUNCTION (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 This trace entry is put out if a call is made to module LDLD1 specifying an invalid function. DATA1 = parameter list				
2	HEX	1801	TRLD1_INVALID_ FUNCTION	

Len	Type	Value	Name	Description
SVC_ EXCEPTION (LDLD functional gate) level = EXCEPTION module = DFHLDLD1/DFHLDDMI Generated whenever a bad return code is received from the SVC service routine which provides Loader authorised facilities. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load)				
2	HEX	1802	TRLD1_SVC_ EXCEPTION	
SVC_ EXCEPTION (LDLD functional gate) level = EXCEPTION module = DFHLDLD2 Generated on return from the Loader's SVC service routine if a bad return code has been presented by the routine. DATA1 = Authorised Facility Plist				
2	HEX	2906	TRLD2_SVC_ EXCEPTION	
SVC_ EXCEPTION (LDLD functional gate) level = EXCEPTION module = DFHLDLD3 Generated on return from the Loader's SVC service routine if a bad return code has been presented by the routine. DATA1 = Authorised Facility Plist < DATA2 > = DESERV GET o/p area (DESB)				
2	HEX	390D	TRLD3_SVC_ EXCEPTION	
MODE CHANGE FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD3 These trace entries are put out whenever a request for change of dispatch mode fails. DATA1 = Call Parameter list.				
2	HEX	390E	TRLD3_MODE_CHANGE	
Long name value input to CONVERT_ NAME (LDLD gate) level = EXCEPTION the convert has just failed module = DFHLDLD3 DATA1 = LDLD_ LONG_NAME parameter input to convert				
2	HEX	3910	TRLD3_LONG_NAME	
CORRUPT CONTROL BLOCK (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 This trace entry is put out whenever a bad control block field is detected. DATA1 = Control block address. DATA2 = Control block identifier. DATA3 = Control block.				
2	HEX	1803	TRLD1_BAD_ STRUCTURE	
LOAD EXCEPTION (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 Generated whenever a CSVQUERY call fails to locate a module in the LPA. DATA1 = Program name DATA1 = Return code				
2	HEX	1804	TRLD1_CSVQUERY_ EXCEPTION	
APE GETMAIN FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 This trace entry is put out whenever a GETMAIN for an APE fails. DATA1 = Call Parameter list.				
2	HEX	1903	TRLD1_APE_GETMAIN	
CSECTL GETMAIN FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 This trace entry is put out whenever a GETMAIN for a CSECTL fails. DATA1 = Call Parameter list.				
2	HEX	1905	TRLD1_CSECTL_ GETMAIN	
PGM GETMAIN FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 This trace entry is put out whenever a GETMAIN for program staorage. DATA1 = Call Parameter list.				
2	HEX	1907	TRLD1_PGM_GETMAIN	
CDE GETMAIN FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 This trace entry is put out whenever a GETMAIN for a dummy CDE fails. DATA1 = Call Parameter list.				
2	HEX	1928	TRLD1_CDE_ GETMAIN_FAIL	
STATE LOCK FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 These trace entries are put out whenever a LOCK request fails for the state lock. DATA1 = Call Parameter list.				
2	HEX	1910	TRLD1_STATE_LOCK	
2	HEX	1911	TRLD1_STATE_LOCK_1	
2	HEX	1912	TRLD1_STATE_LOCK_2	
2	HEX	1913	TRLD1_STATE_LOCK_3	
2	HEX	1914	TRLD1_STATE_LOCK_4	
2	HEX	1902	TRLD1_STATE_LOCK_5	
2	HEX	192D	TRLD1_STATE_LOCK_6	
STATE UNLOCK FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 These trace entries are put out whenever an UNLOCK request fails for the state lock. DATA1 = Call Parameter list.				
2	HEX	1915	TRLD1_STATE_UNLOCK	
2	HEX	1916	TRLD1_STATE_ UNLOCK_1	
2	HEX	1917	TRLD1_STATE_ UNLOCK_2	
2	HEX	1918	TRLD1_STATE_ UNLOCK_3	

Len	Type	Value	Name	Description
2	HEX	1929	TRLD1_STATE_ UNLOCK_4	
LIBRARY LOCK FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 These trace entries are put out whenever a LOCK request fails for the library lock. DATA1 = Call Parameter list.				
2	HEX	1919	TRLD1_LIBRARY_LOCK	
2	HEX	191A	TRLD1_LIBRARY_ LOCK_1	
2	HEX	191B	TRLD1_LIBRARY_ LOCK_2	
2	HEX	192B	TRLD1_LIBRARY_ LOCK_3	
2	HEX	3909	TRLD3_LIBRARY_LOCK	
2	HEX	390E	TRLD3_LIBRARY_ LOCK_1	
LIBRARY UNLOCK FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 These trace entries are put out whenever an UNLOCK request fails for the library lock. DATA1 = Call Parameter list.				
2	HEX	191C	TRLD1_LIBRARY_ UNLOCK	
2	HEX	191D	TRLD1_LIBRARY_ UNLOCK_1	
2	HEX	191E	TRLD1_LIBRARY_ UNLOCK_2	
2	HEX	191F	TRLD1_LIBRARY_ UNLOCK_3	
2	HEX	192C	TRLD1_LIBRARY_ UNLOCK_4	
2	HEX	390A	TRLD3_LIBRARY_ UNLOCK	
2	HEX	390F	TRLD3_LIBRARY_ UNLOCK_1	
MODE CHANGE FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 These trace entries are put out whenever a request for change of dispatch mode fails. DATA1 = Call Parameter list.				
2	HEX	1920	TRLD1_MODE_CHANGE	
2	HEX	1921	TRLD1_MODE_ CHANGE_1	
2	HEX	192A	TRLD1_MODE_ CHANGE_2	
NO OS STORAGE (LDLD functional gate) level = EXCEPTION module = DFHLDLD1 Generated whenever an MVS LOAD or BLDL request fails due to lack of OS storage. DATA1 = Call Parameter list.				
2	HEX	1922	TRLD1_NO_ OS_STORAGE	
2	HEX	1923	TRLD1_NO_ OS_STORAGE_1	
LIBRARY I/O ERROR level = EXCEPTION module = DFHLDLD1 Generated whenever an MVS LOAD or BLDL request fails due to I/O errors on the library. DATA1 = Call Parameter list.				
2	HEX	1924	TRLD1_LIBRARY_ IO_ERROR	
2	HEX	1925	TRLD1_LIBRARY_ IO_ERROR_1	
SVC REQUEST FAILURE level = EXCEPTION module = DFHLDLD1 Generated whenever an MVS LOAD or BLDL request fails due to no specific reason. DATA1 = Call Parameter list.				
2	HEX	1926	TRLD1_SVC_ REQUEST_FAILURE	
2	HEX	1927	TRLD1_SVC_ REQUEST_FAILURE_1	
RECOVERY ENTERED (LDLD functional gate) level = EXCEPTION module = DFHLDLD2 This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data				
2	HEX	2701	TRLD2_RECOVERY_ ENTERED	
FAILED CATALOG WRITE (LDLD functional gate) level = EXCEPTION module = DFHLDLD2 This trace entry is put out if a catalog write request returns a response other than ok. DATA1 = CCCC parameter list. DATA2 = Data to be written.				
2	HEX	2901	TRLD2_CC_WRITE	
2	HEX	2909	TRLD2_CC_WRITE_2	
CATALOG DELETE FAILED (LDLD functional gate) level = EXCEPTION module = DFHLDLD2 This trace entry is put out if a bad response is returned by the catalog when requested to delete a program definition record as part of a Loader DELETE_ PROGRAM request. DATA1 = CCCC parameter list				
2	HEX	2902	TRLD2_CC_DELETE	
CPE GETMAIN FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD2 This trace entry is put out whenever a GETMAIN for a CPE fails. DATA1 = Call Parameter list.				
2	HEX	2903	TRLD2_CPE_GETMAIN	

Len	Type	Value	Name	Description
	RECOVERY ENTERED (LDLD functional gate) level = EXCEPTION module = DFHLDLD3 This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data			
2	HEX	3701	TRLD3_RECOVERY_ ENTERED	
	FAILED CATALOG WRITE (LDLD functional gate) level = EXCEPTION module = DFHLDLD3 This trace entry is put out if a catalog write request returns a response other than ok. DATA1 = CCCC parameter list. DATA2 = Data to be written.			
2	HEX	3901	TRLD3_CC_WRITE	
2	HEX	3905	TRLD3_CC_ WRITE_PDB1	
2	HEX	3906	TRLD3_CC_ WRITE_PDB2	
2	HEX	3907	TRLD3_CC_ WRITE_PDB3	
2	HEX	3908	TRLD3_CC_ WRITE_PDB4	
	LDBE GET FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD3 This trace entry is put out if a GETMAIN for an LDBE fails whilst processing a start browse. DATA1 = parameter list			
2	HEX	3902	TRLD3_LDBE_GETMAIN	
	PRVMOD GETMAIN FAILURE (LDLD functional gate) level = EXCEPTION module = DFHLDLD3 This trace entry is put out if the GETMAIN for PRVMOD fails. DATA1 = parameter list			
2	HEX	3904	TRLD3_PRVMOD_ GETMAIN	
	DOMAIN ENTRY (LDNT SM Notify gate) level = 1 module = DFHLDNT Generated as the first operation on entry to the domain for SM STORAGE_ NOTIFY requests. caller. DATA1 = SMNT Parameter list			
2	HEX	4001	TRNT_ENTRY_TRACE	
	DOMAIN EXIT (LDNT SM Notify gate) level = 1 or EXCEPTION module = DFHLDNT Generated as the final operation prior to performing return via the Kernel to the Loader's caller. DATA1 = SMNT Parameter list			
2	HEX	4002	TRNT_EXIT_TRACE	
	RECOVERY ENTERED (LDNT compression gate) level = EXCEPTION module = DFHLDNT This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data			
2	HEX	4701	TRNT_RECOVERY_ ENTERED	
	INVALID FORMAT (LDNT compression gate) level = EXCEPTION module = DFHLDNT This trace entry is put out if a call is made to the LDNT gate using the incorrect parameter list format DATA1 = parameter list			
2	HEX	4801	TRNT_INVALID_ FORMAT	
	INVALID FUNCTION (LDNT compression gate) level = EXCEPTION module = DFHLDNT This trace entry is put out if a call is made to the LDNT gate specifying an invalid function. DATA1 = parameter list			
2	HEX	4802	TRNT_INVALID_ FUNCTION	
	INVALID PARAMETERS (LDNT compression gate) level = EXCEPTION module = DFHLDNT This trace entry is put out if an invalid combination of parameters is detected. DATA1 = parameter list			
2	HEX	4803	TRNT_INVALID_ PARAMETERS	
	LOCK FAILURE (LDNT compression gate) level = EXCEPTION module = DFHLDNT This trace entry is put out if a request to LOCK the Loader state lock fails. DATA1 = parameter list			
2	HEX	4901	TRNT_LOCK_FAILURE	
	UNLOCK FAILURE (LDNT compression gate) level = EXCEPTION module = DFHLDNT This trace entry is put out if a request to UNLOCK the Loader state lock fails. DATA1 = parameter list			
2	HEX	4902	TRNT_UNLOCK_ FAILURE	
	DOMAIN ENTRY (LDST Statistics gate) level = 1 module = DFHLDST Generated as the first operation on entry to the domain for ST COLLECT_ STATISTICS requests. caller. DATA1 = STST Parameter list			
2	HEX	5001	TRST_ENTRY_TRACE	
	DOMAIN EXIT (LDST Statistics gate) level = 1 or EXCEPTION module = DFHLDST Generated as the final operation prior to performing return via the Kernel to the Loader's caller. DATA1 = STST Parameter list			
2	HEX	5002	TRST_EXIT_TRACE	

Len	Type	Value	Name	Description
	RECOVERY ENTERED (LDST statistics gate) level = EXCEPTION module = DFHLDST This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data			
2	HEX	5701	TRST_RECOVERY_ ENTERED	
	INVALID FORMAT (LDST statistics gate) level = EXCEPTION module = DFHLDST This trace entry is put out if a call is made to the LDST gate using the incorrect parameter list format DATA1 = parameter list			
2	HEX	5801	TRST_INVALID_ FORMAT	
	INVALID FUNCTION (LDST statistics gate) level = EXCEPTION module = DFHLDST This trace entry is put out if a call is made to the LDST gate specifying an invalid function. DATA1 = parameter list			
2	HEX	5802	TRST_INVALID_ FUNCTION	
	INVALID PARAMETERS (LDST statistics gate) level = EXCEPTION module = DFHLDST This trace entry is put out if an invalid combination of parameters is detected. DATA1 = parameter list			
2	HEX	5803	TRST_INVALID_ PARAMETERS	
	LOCK FAILURE (LDST statistics gate) level = EXCEPTION module = DFHLDST This trace entry is put out if a request to LOCK the Loader state lock fails. DATA1 = parameter list			
2	HEX	5901	TRST_LOCK_FAILURE	
	UNLOCK FAILURE (LDST statistics gate) level = EXCEPTION module = DFHLDST This trace entry is put out if a request to UNLOCK the Loader state lock fails. DATA1 = parameter list			
2	HEX	5902	TRST_UNLOCK_ FAILURE	
	DOMAIN ENTRY (LDDM init/term gate) level = 1 module = DFHLDDM Generated as the first operation on entry to the domain for all calls. caller. DATA1 = Domain Manager Parameter list			
2	HEX	6001	TRDM_ENTRY_TRACE	
	DOMAIN EXIT (LDDM init/term gate) level = 1 or EXCEPTION module = DFHLDDM Generated as the final operation prior to performing return via the Kernel to the Loader's caller. DATA1 = Domain Manager Parameter list			
2	HEX	6002	TRDM_EXIT_TRACE	
	PRE-SVC CALL (LDLD functional gate) level = 2 module = DFHLDDM Generated immediately prior to issuing an SVC to invoke the Loader's authorised facilities module. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load)			
2	HEX	6003	TRDM_SVC_CALL	
	POST SVC-CALL (LDLD functional gate) level = 2 module = DFHLDDM Generated on return from the Loader's SVC service routine if a normal return code has been presented by the routine. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load)			
2	HEX	6004	TRDM_SVC_RETURN	
	POST SVC-CALL (LDLD functional gate) level = EXCEPTION module = DFHLDDM Generated whenever a bad return code is received from the SVC service routine which provides Loader authorised facilities. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load)			
2	HEX	6005	TRDM_SVC_EXCEPTION	
	RECOVERY ENTERED (LDDM service gate) level = EXCEPTION module = DFHLDDM This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data			
2	HEX	6701	TRDM_RECOVERY_ ENTERED	
	INVALID FORMAT (LDDM service gate) level = EXCEPTION module = DFHLDDM This trace entry is put out if a call is made to the LDDM gate using the incorrect parameter list format DATA1 = parameter list			
2	HEX	6801	TRDM_INVALID_ FORMAT	
	INVALID FUNCTION (LDDM service gate) level = EXCEPTION module = DFHLDDM This trace entry is put out if a call is made to the LDLD gate specifying an invalid function. DATA1 = parameter list			
2	HEX	6802	TRDM_INVALID_ FUNCTION	

Len	Type	Value	Name	Description
	INVALID PARAMETERS (LDDM service gate) level = EXCEPTION module = DFHLDDM This trace entry is put out if an invalid combination of parameters is detected. DATA1 = parameter list			
2	HEX	6803	TRDM_INVALID_ PARAMETERS	
	BAD LOB READ FROM CATALOG (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad fields is detected in the Loader Option Block (LOB) read from the catalog during pre-initialisation. DATA1 = LOB			
2	HEX	6804	TRDM_BAD_CC_LOB	
	DEFINE PROGRAM (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is encountered whilst defining the Loaders secondary initialisation module DFHLDDMI. DATA1 = parameter list.			
2	HEX	6901	TRDM_DEFINE	
	ACQUIRE PROGRAM (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is encountered whilst acquiring the Loaders secondary initialisation module DFHLDDMI. DATA1 = parameter list.			
2	HEX	6902	TRDM_ACQUIRE	
	RELEASE PROGRAM (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is encountered whilst releasing the Loaders secondary initialisation module DFHLDDMI. DATA1 = parameter list.			
2	HEX	6903	TRDM_RELEASE	
	GETMAIN (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is encountered whilst acquiring the staorage for the Loaders global storage. DATA1 = parameter list.			
2	HEX	6905	TRDM_GETMAIN	
	ADD GATE (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is encountered whilst adding the LDLD gate. DATA1 = parameter list.			
2	HEX	6908	TRDM_ADD_GATE	
	GET PARMS (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is encountered whilst requesting start-up override parameters. DATA1 = parameter list.			
2	HEX	6909	TRDM_GET_PARMS	
	CC WRITE (LDDM initialisation) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is encountered whilst writing out the LOB during quiesce. DATA1 = parameter list.			
2	HEX	690B	TRDM_CC_WRITE	
	ADD SUBPOOL (LDDM initialisation) level = EXCEPTION module = DFHLDDM Thes trace entries are put out if a bad response is encountered whilst adding one of the Loaders storage subpools. DATA1 = parameter list.			
2	HEX	690D	TRDM_ADD_ CONTROL_POOL_FAIL	
2	HEX	690E	TRDM_ADD_ APE_CELL_POOL_ FAIL	
2	HEX	6923	TRDM_ADD_ CPE_POOL_FAIL	
2	HEX	690F	TRDM_ADD_ CSECTL_POOL_FAIL	
2	HEX	6910	TRDM_ADD_ LDNUC_POOL_FAIL	
2	HEX	6911	TRDM_ADD_ LDENUC_POOL_FAIL	
2	HEX	6922	TRDM_ADD_ LDNUCRO_POOL_FAIL	
2	HEX	6912	TRDM_ADD_ LDENUCRO_POOL_ FAIL	
2	HEX	6913	TRDM_ADD_ LDRES_POOL_FAIL	
2	HEX	6914	TRDM_ADD_ LDERES_POOL_FAIL	
2	HEX	6920	TRDM_ADD_ LDRESRO_POOL_FAIL	
2	HEX	6915	TRDM_ADD_ LDERESRO_POOL_ FAIL	
2	HEX	6916	TRDM_ADD_ LDPGM_POOL_FAIL	
2	HEX	6917	TRDM_ADD_ LDEPGM_POOL_FAIL	

Len	Type	Value	Name	Description
2	HEX	6921	TRDM_ADD_LDPGMRO_POOL_FAIL	
2	HEX	6918	TRDM_ADD_LDEPGMRO_POOL_ FAIL	
2	HEX	6924	TRDM_ADD_CDE_POOL_FAIL	
2	HEX	6925	TRDM_ADD_LDNRS_POOL_FAIL	
2	HEX	6926	TRDM_ADD_LDENRS_POOL_FAIL	
2	HEX	6927	TRDM_ADD_LDNRSRO_POOL_FAIL	
2	HEX	6928	TRDM_ADD_LDENRSRO_POOL_ FAIL	
				SET ANCHOR (LDDM initialisation) level = EXCEPTION module = DFHLDDM These trace entries are put out if a bad response is encountered whilst defining the Loaders global storage to the Kernel. DATA1 = parameter list.
2	HEX	6919	TRDM_SET_ANCHOR	
2	HEX	691A	TRDM_SET_ANCHOR_1	
				ADD LOCK (LDDM initialisation) level = EXCEPTION module = DFHLDDM These trace entries are put out if a bad response is encountered whilst adding one of the Loaders locks. DATA1 = parameter list.
2	HEX	691B	TRDM_ADD_LOCK	
2	HEX	691C	TRDM_ADD_LOCK_1	
				UNLOCK (LDDM initialisation) level = EXCEPTION module = DFHLDDM These trace entries are put out if a bad response is encountered whilst UNLOCKing one of the Loader locks. DATA1 = parameter list.
2	HEX	691D	TRDM_UNLOCK	
2	HEX	691E	TRDM_UNLOCK_1	
				INQUIRE START (LDDM pre-initialise) level = EXCEPTION module = DFHLDDM This trace entry is put out if a bad response is returned when we check whether this CICS startup is cold or not. (using INQUIRE_ START). DATA1 = DMDM parameter list. DATA2 = PAGP parameter list
2	HEX	691F	TRDM_INQUIRE_START	
				PRE-SVC CALL (LDDM initialisation) level = 1 module = DFHLDDMI Generated immediately prior to issuing an SVC to invoke the Loader's authorised facilities module. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load) Up to the first 200 characters
2	HEX	7003	TRDMI_SVC_CALL	
				POST SVC-CALL (LDDM initialisation) level = 1 module = DFHLDDMI Generated on return from the Loader's SVC service routine if a normal return code has been presented by the routine. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load) Up to the first 200 characters
2	HEX	7004	TRDMI_SVC_RETURN	
				PRE-LOAD (initialisation) level = 1 module = DFHLDDMI Generated prior to issuing a CSVQUERY call to access an LPA resident module. DATA1 = Program name
2	HEX	7005	TRDMI_PRE_CSVQUERY	
				POST LOAD (initialisation) level = 1 module = DFHLDDMI Generated after issuing a CSVQUERY call to access an LPA resident module. DATA1 = Program name DATA2 = Return code
2	HEX	7006	TRDMI_POST_ CSVQUERY	
				RECOVERY ENTERED (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out if the active recovery routine is driven. DATA1 = parameter list DATA2 = Kernel error data
2	HEX	7701	TRDMI_RECOVERY_ENTERED	
				SVC EXCEPTION (initialisation) level = EXCEPTION module = DFHLDDMI Generated whenever a bad return code is received from the SVC service routine which provides Loader authorised facilities. DATA1 = Authorised Facility Plist < DATA2 > = BLDL Plist (BLDL or RPL load) Up to the first 200 characters
2	HEX	7801	TRDMI_SVC_ EXCEPTION	
				INVALID PDB (LDDMI init rtne) level = EXCEPTION module = DFHLDDMI This trace entry is put out if an invalid PDB is detected. DATA1 = program name DATA2 = PDB

Len	Type	Value	Name	Description
2	HEX	7802	TRDMI_BAD_PDB	
				LOAD EXCEPTION (initialisation) level = EXCEPTION module = DFHLDDMI Generated whenever a CSVQUERY call fails to locate a module in the LPA. DATA1 = Program name DATA1 = Return code
2	HEX	7803	TRDMI_CSVQUERY_ EXCEPTION	
				LOAD EXCEPTION (initialisation) level = EXCEPTION module = DFHLDDMI Generated when a CSVQUERY call fails when attempting to locate DFHSIP. DATA1 = Program name DATA1 = Return code
2	HEX	7832	TRDMI_DFHSIP_ NOT_FOUND	
				GET PARMS (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out if a bad response is encountered whilst requesting start-up override parameters. DATA1 = parameter list.
2	HEX	7903	TRDMI_GET_PARMS	
				APE GETMAIN FAILURE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a GETMAIN for an APE fails. DATA1 = Call Parameter list.
2	HEX	7905	TRDMI_APE_GETMAIN	
				WAIT PHASE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out when the request to wait for the global catalog fails DATA1 = Call Parameter list.
2	HEX	7906	TRDMI_WAIT_PHASE	
				LOCAL CATALOG (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a LOCAL catalog request fails. DATA1 = Call Parameter list.
2	HEX	7907	TRDMI_LOCAL_ CATALOG	
				GLOBAL CATALOG (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a GLOBAL catalog request fails. DATA1 = Call Parameter list.
2	HEX	7908	TRDMI_GLOBAL_ CATALOG	
				DFHLDNT (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out when a problem is encountered in establishing the SMNT gate or in defining program DFHLDNT DATA1 = Call Parameter list.
2	HEX	7909	TRDMI_DFHLDNT	
				DFHLDST (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out when a problem is encountered in establishing the STST gate or in defining program DFHLDST DATA1 = Call Parameter list.
2	HEX	790A	TRDMI_DFHLDST	
				LIBRARY LOCK FAILURE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a LOCK request fails for the library lock. DATA1 = Call Parameter list.
2	HEX	790B	TRDMI_LIBRARY_LOCK	
				LIBRARY UNLOCK FAILURE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever an UNLOCK request fails for the library lock. DATA1 = Call Parameter list.
2	HEX	790C	TRDMI_LIBRARY_ UNLOCK	
2	HEX	7935	TRDMI_LIBRARY_ UNLOCK_2	
				START BROWSE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a START_ BROWSE request fails. DATA1 = Call Parameter list.
2	HEX	790D	TRDMI_START_BROWSE	
				END BROWSE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever an END_ BROWSE request fails. DATA1 = Call Parameter list.
2	HEX	790E	TRDMI_END_BROWSE	
				CPE GETMAIN FAILURE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a GETMAIN for a CPE fails. DATA1 = Call Parameter list.
2	HEX	790F	TRDMI_CPE_GETMAIN	
				BLDL GETMAIN FAILURE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a GETMAIN for a BLDL plist fails. DATA1 = Call Parameter list.
2	HEX	7910	TRDMI_BLDL_GETMAIN	

Len	Type	Value	Name	Description
CSECTL GETMAIN FAILURE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a GETMAIN for a CSECTL fails. DATA1 = Call Parameter list.				
2	HEX	7912	TRDMI_CSECTL_ GETMAIN	
MODE CHANGE FAILURE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out whenever a request for change of dispatch mode fails. DATA1 = Call Parameter list.				
2	HEX	7913	TRDMI_MODE_CHANGE	
INQUIRE START (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out if a bad response is encountered whilst requesting value of START= SIT parameter. DATA1 = parameter list.				
2	HEX	7914	TRDMI_INQUIRE_ START	
TYPE PURGE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out if a bad response is encountered whilst attempting a TYPE_ PURGE to the Catalog domain. DATA1 = parameter list.				
2	HEX	7915	TRDMI_TYPE_PURGE	
STATE LOCK (LDDM initialisation) level = EXCEPTION module = DFHLDDMI These trace entries are put out if a bad response is encountered whilst LOCKing the Loader state lock. DATA1 = parameter list.				
2	HEX	7920	TRDMI_STATE_LOCK	
2	HEX	7921	TRDMI_STATE_LOCK_1	
2	HEX	7922	TRDMI_STATE_LOCK_2	
2	HEX	7923	TRDMI_STATE_LOCK_3	
2	HEX	7924	TRDMI_STATE_LOCK_4	
2	HEX	7925	TRDMI_STATE_LOCK_5	
2	HEX	7932	TRDMI_STATE_LOCK_6	
STATE UNLOCK (LDDM initialisation) level = EXCEPTION module = DFHLDDMI These trace entries are put out if a bad response is encountered whilst UNLOCKing the Loader state lock. DATA1 = parameter list.				
2	HEX	7926	TRDMI_STATE_UNLOCK	
2	HEX	7927	TRDMI_STATE_ UNLOCK_1	
2	HEX	7928	TRDMI_STATE_ UNLOCK_2	
2	HEX	7929	TRDMI_STATE_ UNLOCK_3	
2	HEX	792A	TRDMI_STATE_ UNLOCK_4	
2	HEX	792B	TRDMI_STATE_ UNLOCK_5	
2	HEX	792C	TRDMI_STATE_ UNLOCK_6	
2	HEX	792D	TRDMI_STATE_ UNLOCK_7	
2	HEX	792E	TRDMI_STATE_ UNLOCK_8	
2	HEX	7933	TRDMI_STATE_ UNLOCK_9	
ADD GATE (LDDM initialisation) level = EXCEPTION module = DFHLDDMI These trace entries are put out if a bad response is encountered whilst adding the LDLD gate. DATA1 = parameter list.				
2	HEX	7930	TRDMI_ADD_GATE	
2	HEX	7931	TRDMI_ADD_GATE_1	
DISPATCHER CALL FAILURES (LDDM initialisation) level = EXCEPTION module = DFHLDDMI These trace entries are put out if a bad response is returned from DSSR SUSPEND, DSSR ADD_ SUSPEND and DSSR DELETE_ SUSPEND. DATA1 = parameter list.				
2	HEX	7934	TRDMI_ADD_SUSPEND	
2	HEX	7938	TRDMI_SUSPEND_FAIL	
2	HEX	7936	TRDMI_DELETE_ SUSPEND_FAIL	
SMGF GETMAIN (LDDM initialisation) level = EXCEPTION module = DFHLDDMI This trace entry is put out if a bad response is returned from SMGF GETMAIN when attempting to getmain a LDWE. DATA1 = parameter list.				
2	HEX	7937	TRDMI_LDWE_GETMAIN	

LGANC Logger domain anchor block

-

This anchor block contains the global storage for the LG domain.
It is divided into two distinct halves, one half for DFHLGxx
modules and one half for DFHL2xx modules.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	2048	LGA	
(0)	CHARACTER	1024	LGA_LG_PART	
- Block Header				
(0)	CHARACTER	16	LGA_PREFIX	===> eyecatcher <===
(0)	HALFWORD	2	LGA_LENGTH	length of lga
(2)	CHARACTER	14	LGA_PREFIX_ TEXT	>DFHLGAnchor
-- - Domain state information				
(10)	ADDRESS	4	LGA_LOCK_TOKEN	LG domain lock token
(14)	UNSIGNED	1	LGA_LG_STATE	LG domain state initialised, quiesced or terminated
(15)	UNSIGNED	1	LGA_FLAGS	
	1...		LGA_COLD_START	1=CICS cold started
	.1..		LGA_INITIAL_ START	1=CICS initial start
	..1.		LGA_XLGSTRM_ ACTIVE	1=XLGSTRM exit active
	...1		LGA_XLGWBC_ ACTIVE	1=XLGWBC exit active
	.... 1...		LGA_XRSINDI_ ACTIVE	1=XRSINDI exit active
(16)	CHARACTER	2	*	
-- - Subpool Tokens				
(18)	CHARACTER	8	LGA_GENERAL_ SPTOKEN	token received when lga was GETMAINed
(20)	CHARACTER	8	LGA_SD_SUBPOOL_TOKEN	Token for Stream Data entries subpool
(28)	CHARACTER	8	LGA_GD_SUBPOOL_TOKEN	Token for Glog Data entries subpool
(30)	CHARACTER	8	LGA_JI_SUBPOOL_TOKEN	Token for Journal entries subpool
(38)	CHARACTER	8	LGA_JM_SUBPOOL_TOKEN	Token for JournalModel entries subpool
(40)	CHARACTER	8	LGA_BR_SUBPOOL_TOKEN	Token for browse token entries subpool
(48)	CHARACTER	8	LGA_UW_SUBPOOL_TOKEN	Token for Unit of Work entries subpool
-- - Pointers				
(50)	ADDRESS	4	LGA_SD_HDR_PTR	-> Stream data header
(54)	ADDRESS	4	LGA_GD_HDR_PTR	-> Glog data header
(58)	ADDRESS	4	LGA_JI_HDR_PTR	-> Journal info header
(5C)	ADDRESS	4	LGA_JM_HDR_PTR	-> JournalModel data header
(60)	ADDRESS	4	LGA_BR_HDR_PTR	-> Browse data header

Offset Hex	Type	Len	Name (Dim)	Description
--				
-				
Statistics				
(64)	ADDRESS	4	LG_STATS_ BUFFER_PTR	Statistics buffer
(68)	CHARACTER	8	LGA_LAST_JNL_RESET_TIME	jnl stats last reset@L7A
(70)	CHARACTER	8	LGA_LAST_LSN_RESET_TIME	lsn stats last reset@L7A
--				
-				
Misc fields				
(78)	ADDRESS	4	LGA_JN_ENQPOOL_TOKEN	Journal Enqueue pool
(7C)	ADDRESS	4	LGA_ST_ENQPOOL_TOKEN	Streamname Enqueue pool
(80)	ADDRESS	4	LGA_SMF_ LOCK_TOKEN	Shared SMF jnl lock
(84)	CHARACTER	9	LGA_USERID	Jobstep userid
(84)	UNSIGNED	1	LGA_USERID_L	length
(85)	CHARACTER	8	LGA_USERID_N	value
(8D)	CHARACTER	9	LGA_APPLID	Generic applid
(8D)	UNSIGNED	1	LGA_APPLID_L	length
(8E)	CHARACTER	8	LGA_APPLID_N	value
(96)	BITSTRING	1	LGA_L2_FLAGS	L2 flags
	1...		LGA_L2_ACTIVE	L2 activated
(97)	CHARACTER	1	*	reserved
(98)	ADDRESS	4	LGA_LGUOW_LOCK_TOKEN	Lock for browsing UOW chain
(9C)	CHARACTER	5	LGA_SYSID	Sysid
(9C)	UNSIGNED	1	LGA_SYSID_L	length
(9D)	CHARACTER	4	LGA_SYSID_N	value
(A1)	CHARACTER	11	*	reserved
--				
(400)	CHARACTER	1024	LGA_L2_PART	
-				
This portion of the Log Manager anchor block is for the exclusive use of the DFHL2xx modules. The data is owned by DFHL2DM and is mapped by copybook DFHL2xxC.				
(400)	CHARACTER	1024	*	
--				
(800)	CHARACTER		LGA_END	

-

Stream data represents the state of a single MVS log stream.

The entire set of MVS log streams is stored as an AVL tree structure. The tree header and element leaf pointers are maintained by the BB/LX building block and are not mapped here

The stream data tree is maintained by DFHLGST but some other routines within the logger domain do modify individual stream data entries.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	LGSD_STREAM_DATA	
(0)	CHARACTER	26	LGSD_STREAM	MVS log stream name
(1A)	UNSIGNED	1	LGSD_SYSTEM_LOG	Is log a system log? 1=Yes, 2=No
(1B)	UNSIGNED	1	LGSD_FAILED_LOG	Has stream failed 1=Yes, 2=No
(1C)	FULLWORD	4	LGSD_USE_CT	Count of users of stream
(20)	ADDRESS	4	LGSD_STREAM_LOCK	Stream lock token
(24)	ADDRESS	4	LGSD_LOGBUF_TKN	-> Buffers etc.

Offset Hex	Type	Len	Name (Dim)	Description
(28)	CHARACTER	16	LGSD_STRUCTURE_ NAME	MVS LS structure name

--
-

The data retained for each explicitly opened general log.

A storage block table (pointed to by lga_gd_hdr_ptr)
contains pointers to each glog_data entry

The glog data is processed solely by DFHLGGL

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	36	LGGD_GLOG_DATA	
(0)	ADDRESS	4	LGGD_LOG_TOKEN	Log token for this block
(4)	CHARACTER	8	LGGD_USER_TOKEN	Opener's reference
(C)	ADDRESS	4	LGGD_STREAM_ TOKEN	Log stream token for MVS Logbuf token for SMF
(10)	CHARACTER	8	LGGD_JNAME	Journal name
(18)	CHARACTER	2	LGGD_COMPONENT	Component identifier
(1A)	UNSIGNED	1	LGGD_LOGTYPE	1=Mvs, 2=Smf, 3=Dummy
(1B)	CHARACTER	1	*	Reserved
(1C)	FULLWORD	4	LGGD_DOMAIN_NO	Domain opening log
(20)	FULLWORD	4	LGGD_ERROR_GATE	Gate# for error callback

--
-

Journal Info represents the state of a single CICS user journal.

The entire set of Journals is stored as an AVL tree structure. The
tree header and element leaf pointers are maintained by the BB/LX
building block and are not mapped here

The journal info tree is used only by DFHLGJN.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	60	LGJI_JOURNAL_INFO	
(0)	CHARACTER	8	LGJI_JNAME	Journal name
(8)	CHARACTER	26	LGJI_STREAM	MVS log stream name
(22)	UNSIGNED	1	LGJI_LOG_TYPE	1=Mvs, 2=Smf, 3=Dummy
(23)	UNSIGNED	1	LGJI_SYSTEM_LOG	Is jnl a system log? 1=Yes, 2=No
(24)	UNSIGNED	1	LGJI_STATUS	Journal status 1=Connected 2=Disconnected 3=Disabled 5=Failed
(25)	UNSIGNED	1	LGJI_FAIL_REASON	Failure reason code (same as lgjn_reason) 6=unable_to_create_jnl 7=system_log_conflict 9=jnl_has_failed 10=error_opening_log 11=write_error
(26)	CHARACTER	2	*	
(28)	ADDRESS	4	LGJI_STREAM_ TOKEN	Log stream token Logbuf token for SMF
(2C)	FULLWORD	4	LGJI_JNLWRITE_ COUNT	Stats - write count
(30)	BITSTRING	8	LGJI_JNLWRITE_ BYTES	- bytes total
(38)	FULLWORD	4	LGJI_JNLFLUSH_ REQS	- flushes

--
-

The data retained for each browse of a log manager resource.

A storage block table (pointed to by lga_br_hdr_ptr)
contains pointers to each browse_data entry

The Browse data is used for all browses in DFHLGST, DFHLGJN,
DFHLGLD

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	31	LGBR_BROWSE_DATA	
(0)	ADDRESS	4	LGBR_BROWSE_ TOKEN	Token for this block
(4)	UNSIGNED	1	LGBR_TYPE	Resource type
(5)	CHARACTER	26	LGBR_KEY	Browse key
(5)	CHARACTER	8	LGBR_JNAME	Journal name
(5)	CHARACTER	8	LGBR_JMNAME	JournalModel name
(5)	CHARACTER	26	LGBR_STREAM	Stream name

--
-

JournalModel content represents a single installed JournalModel resource.

The set of installed JournalModels are maintained on the global catalog. In storage they are maintained as a linked list.

NOTE: Templates names are stored in an internal format where values.

The JournalModel content is used only by DFHLGLD

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	LGJMC_JOURNALMODEL_ CONTENT	
(0)	CHARACTER	8	LGJMC_JOURNALMODEL_ NAME	JournalModel name
(8)	CHARACTER	8	LGJMC_JNL_ TEMPLATE_X	Jnl template-extnl format
(10)	CHARACTER	8	LGJMC_JNL_ TEMPLATE_I	Jnl template-intnl format
(18)	CHARACTER	26	LGJMC_STREAM_ PROTO	Prototype Log stream name
(32)	UNSIGNED	1	LGJMC_LOG_TYPE	1=Mvs, 2=Smf, 3=Dummy
(33)	CHARACTER	1	*	

--
-

The data retained for each unit of work that has written log reords with the Force_ at_sync option

The data is maintained as a simple linked list anchored in the uow_token.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	LGUOW_HEADER	Work unit header
(0)	ADDRESS	4	LGUOW_CHAIN_HEAD	Chain header
(4)	CHARACTER	8	LGUOW_TIME_STAMP	Time of first log write

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	LGUOW_STREAM_ FORCE	Streams used
(0)	ADDRESS	4	LGUOW_CHAIN_NEXT	Chain link
(4)	ADDRESS	4	LGUOW_STREAM_ TOKEN	
(8)	ADDRESS	4	LGUOW_FORCE_ TOKEN	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	510	LGGD_BLOCKING	no. of entries/block in the storage table hdr
4	DECIMAL	20	LGBR_BLOCKING	no of entries/block in the storage table hdr

--
-

Constants

-

LG Domain States (printed in formatted dump)

1	DECIMAL	1	LG_STATE_ INITIALISING
1	DECIMAL	2	LG_STATE_ INITIALISED
1	DECIMAL	3	LG_STATE_QUIESCING
1	DECIMAL	4	LG_STATE_QUIESCED
1	DECIMAL	5	LG_STATE_ TERMINATED

--
-

Log manager message numbers and system dumpcode values

4	DECIMAL	1	MNO_ABEND
8	CHARACTER	LG0001	DCD_ABEND
4	DECIMAL	2	MNO_SEVERE_ERROR
8	CHARACTER	LG0002	DCD_SEVERE_ERROR
4	DECIMAL	3	MNO_NO_STORAGE
8	CHARACTER	LG0003	DCD_NO_STORAGE
4	DECIMAL	101	MNO_DOM_INIT_START
4	DECIMAL	102	MNO_DOM_INIT_END
4	DECIMAL	301	MNO_JNL_FAILED
4	DECIMAL	302	MNO_JNL_DEFINED
4	DECIMAL	303	MNO_JNL_CONN_FAIL
4	DECIMAL	304	MNO_JNL_CATLG_FAIL
4	DECIMAL	305	MNO_JNL_CATLG_ DEL_FAIL
4	DECIMAL	306	MNO_JNL_DISCARDED
4	DECIMAL	401	MNO_JOURNALMODEL_INSTALLED
4	DECIMAL	402	MNO_JOURNALMODEL_CATLG_FAIL
4	DECIMAL	403	MNO_JOURNALMODEL_CATLG_DEL_FAIL
4	DECIMAL	404	MNO_JOURNALMODEL_REPLACED
4	DECIMAL	405	MNO_JOURNALMODEL_DISCARDED
4	DECIMAL	501	MNO_EXIT_REJECTED_DEFINE
4	DECIMAL	502	MNO_STREAM_DEFINED
4	DECIMAL	503	MNO_STREAM_DEFINE_ERROR
8	CHARACTER	LG0503	DCD_STREAM_DEFINE_ERROR
4	DECIMAL	504	MNO_STREAM_DEFINE_NOAUTH
4	DECIMAL	505	MNO_STREAM_DEFINE_BADHLQ
4	DECIMAL	506	MNO_STREAM_DEFINE_INVSPACE
4	DECIMAL	507	MNO_STREAM_DEFINE_MAXSTREAM
4	DECIMAL	508	MNO_STREAM_DEFINE_LIKE
4	DECIMAL	509	MNO_STREAM_DEFINE_STRUCTNAME
4	DECIMAL	510	MNO_STREAM_DEFINE_STREAMNAME
4	DECIMAL	511	MNO_STREAM_DEFINE_NOSTRUCTNAME
4	DECIMAL	512	MNO_STREAM_CONN_CONFLICT
4	DECIMAL	513	MNO_STREAM_CONN_FAILED
4	DECIMAL	514	MNO_STREAM_ENQ_CONFLICT

--
-

Statistics

4	DECIMAL	4096	LG_STATS_ BUFFER_SIZE

Len	Type	Value	Name	Description

--
-

Literals

2	CHARACTER	LG	COMPID	Domain id
8	CHARACTER	LGGENRAL	SPNAME_GENERAL	General purpose subpool for LG domain
14	CHARACTER	>DFHLGANCHOR	LGA_EYE_CATCHER	
8	CHARACTER	ANCHOR	LGA_BLOCKNAME	
8	CHARACTER	STATSBUF	LGA_STATSBUFFER	
8	CHARACTER	LGLOCK	LG_LOCK_NAME	Domain lock
8	CHARACTER	LGSTLOCK	LG_STREAM_LOCK_NAME	Stream lock
8	CHARACTER	LGUOWLCK	LG_LGUOW_LOCK_NAME	UOW lock
8	CHARACTER	DFHLGLOG	LG_LOGOFLOG	Log of logs

--
-

Error codes (for DFHKERN RECOVERY_REQUEST)

4	CHARACTER	ALGA	LOCK_ERROR_CODE	
4	CHARACTER	ALGB	UNLOCK_ERROR_CODE	
4	CHARACTER	ALGC	BBLX_ERROR_CODE	
4	CHARACTER	ALGD	BBLX_SIF_ERROR_CODE	
4	CHARACTER	ALGE	LDMATCH_ERROR_CODE	
4	CHARACTER	ALGF	ENQ_DEQ_ERROR_CODE	
4	CHARACTER	ALGG	CSQC_ERROR_CODE	

--
--
-

Trace Point Identifiers

-

lgdm tracepoints

2	HEX	0101	TID_LGDM_ENTRY	
2	HEX	0102	TID_LGDM_EXIT	
2	HEX	0103	TID_LGDM_RECOVERY	
2	HEX	0104	TID_LGDM_INVALID_FORMAT	
2	HEX	0105	TID_LGDM_INVALID_FUNCTION	
2	HEX	0106	TID_LGDM_RELEASE_LOCK_ERROR	
2	HEX	0107	TID_LGDM_NO_STORAGE_FOR_LGA	
2	HEX	0108	TID_LGDM_REGISTER_ERROR	
2	HEX	0109	TID_LGDM_SET_GATE_ERROR	
2	HEX	0110	TID_LGDM_INVALID_EXIT_ID	
2	HEX	0111	TID_LGDM_GET_PARAMETERS_FAILED	
2	HEX	0112	TID_LGDM_RELEASE_LGUOW_ERROR	

--
-

lggl tracepoints

2	HEX	0201	TID_LGGL_ENTRY	
2	HEX	0202	TID_LGGL_EXIT	
2	HEX	0203	TID_LGGL_RECOVERY	
2	HEX	0204	TID_LGGL_INVALID_FORMAT	
2	HEX	0205	TID_LGGL_INVALID_FUNCTION	
2	HEX	0206	TID_LGGL_UNKNOWN_KE_ERROR_CODE	
2	HEX	0207	TID_LGGL_GET_EXC_LOCK_ERROR	
2	HEX	0208	TID_LGGL_RELEASE_EXC_LOCK_ERROR	
2	HEX	0209	TID_LGGL_GET_SHR_LOCK_ERROR	
2	HEX	020A	TID_LGGL_RELEASE_SHR_LOCK_ERROR	

Len	Type	Value	Name	Description
2	HEX	020B	TID_LGGL_RECOVERY_RELEASE_LOCK_ERROR	
2	HEX	020C	TID_LGGL_ADD_SUBPOOL_ERROR	
2	HEX	020D	TID_LGGL_UNKNOWN_LOG_TOKEN	
2	HEX	020E	TID_LGGL_ BAD_LOGTYPE	
2	HEX	0211	TID_LGGL_GET_SHR_STREAM_LOCK_ERROR	
2	HEX	0212	TID_LGGL_RELEASE_SHR_STREAM_LOCK_ERROR	
2	HEX	0213	TID_LGGL_REC_RLSE_STREAM_LOCK_ERROR	
2	HEX	0214	TID_LGGL_INVALID_PARAMETERS	
2	HEX	0215	TID_LGGL_GLOGS_BBLX_EXCEPTION	
2	HEX	0216	TID_LGGL_GLOGS_SIF_EXCEPTION	
2	HEX	0217	TID_LGGL_ADD_UW_SUBPOOL_ERROR	
2	HEX	0218	TID_LGGL_STORAGE_REQ_PURGED	
2	HEX	0219	TID_LGGL_START_WT_BROWSE_ERROR	
2	HEX	0220	TID_LGGL_GET_NEXT_WT_ERROR	
2	HEX	022A	TID_LGGL_END_WT_BROWSE_ ERROR	
2	HEX	022B	TID_LGGL_MVS_WRITE_ERROR	
2	HEX	022C	TID_LGGL_SMF_WRITE_ERROR	
2	HEX	022D	TID_LGGL_MVS_FORCE_ERROR	
2	HEX	022E	TID_LGGL_SMF_FORCE_ERROR	
2	HEX	0231	TID_LGGL_GET_SHR_SMF_LOCK_ERROR	
2	HEX	0232	TID_LGGL_RELEASE_SHR_SMF_LOCK_ERROR	
2	HEX	0233	TID_LGGL_REC_RLSE_SMF_LOCK_ERROR	
2	HEX	0234	TID_LGGL_GET_EXC_LGUOW_LOCK_ERROR	
2	HEX	0235	TID_LGGL_RELEASE_EXC_LGUOW_LOCK_ERROR	
2	HEX	0236	TID_LGGL_REC_RLSE_LGUOW_LOCK_ERROR	

--
 -

lgjn tracepoints

Len	Type	Value	Name	Description
2	HEX	0301	TID_LGJN_ENTRY	
2	HEX	0302	TID_LGJN_EXIT	
2	HEX	0303	TID_LGJN_RECOVERY	
2	HEX	0304	TID_LGJN_INVALID_FORMAT	
2	HEX	0305	TID_LGJN_INVALID_FUNCTION	
2	HEX	0306	TID_LGJN_UNKNOWN_KE_ERROR_CODE	
2	HEX	0307	TID_LGJN_GET_EXC_LOCK_ERROR	
2	HEX	0308	TID_LGJN_RELEASE_EXC_LOCK_ERROR	
2	HEX	0309	TID_LGJN_GET_SHR_LOCK_ERROR	
2	HEX	030A	TID_LGJN_RELEASE_SHR_LOCK_ERROR	

Len	**Type**	**Value**	**Name**	**Description**
2	HEX	030B	TID_LGJN_RECOVERY_RELEASE_LOCK_ERROR	
2	HEX	030C	TID_LGJN_ADD_SUBPOOL_ERROR	
2	HEX	030D	TID_LGJN_JOURNALS_BBLX_EXCEPTION	
2	HEX	030E	TID_LGJN_JOURNALS_SIF_EXCEPTION	
2	HEX	030F	TID_LGJN_BROWSES_BBLX_EXCEPTION	
2	HEX	0310	TID_LGJN_BROWSES_SIF_EXCEPTION	
2	HEX	0311	TID_LGJN_GET_SHR_STREAM_LOCK_ERROR	
2	HEX	0313	TID_LGJN_REC_RLSE_STREAM_LOCK_ERROR	
2	HEX	0314	TID_LGJN_ JNL_DEFINED	
2	HEX	0315	TID_LGJN_ STREAM_FAILED	
2	HEX	0316	TID_LGJN_INVALID_JNL_STATUS	
2	HEX	0317	TID_LGJN_LD_MATCH_ERROR	
2	HEX	0318	TID_LGJN_INVALID_SET_STATUS	
2	HEX	0319	TID_LGJN_CATLG_WRITE_ERROR	
2	HEX	0320	TID_LGJN_CATLG_DELETE_ERROR	
2	HEX	0321	TID_LGJN_JNL_CONN_ERROR	
2	HEX	0322	TID_LGJN_ENQUEUE_ERROR	
2	HEX	0323	TID_LGJN_DEQUEUE_ERROR	
2	HEX	0324	TID_LGJN_ADD_ENQPOOL_ERROR	
2	HEX	0325	TID_LGJN_ JNL_DISCARDED	
2	HEX	0326	TID_LGJN_GET_SHR_SMF_LOCK_ERROR	
2	HEX	0327	TID_LGJN_GET_EXC_SMF_LOCK_ERROR	
2	HEX	0328	TID_LGJN_RELEASE_EXC_SMF_LOCK_ERROR	
2	HEX	0329	TID_LGJN_REC_RLSE_SMF_LOCK_ERROR	
2	HEX	032A	TID_LGJN_SMF_CONN_ERROR	

--
-

lgld tracepoints

2	HEX	0401	TID_LGLD_ENTRY	
2	HEX	0402	TID_LGLD_EXIT	
2	HEX	0403	TID_LGLD_RECOVERY	
2	HEX	0404	TID_LGLD_INVALID_FORMAT	
2	HEX	0405	TID_LGLD_INVALID_FUNCTION	
2	HEX	0406	TID_LGLD_UNKNOWN_KE_ERROR_CODE	
2	HEX	0407	TID_LGLD_GET_EXC_LOCK_ERROR	
2	HEX	0408	TID_LGLD_RELEASE_EXC_LOCK_ERROR	
2	HEX	0409	TID_LGLD_GET_SHR_LOCK_ERROR	
2	HEX	040A	TID_LGLD_RELEASE_SHR_LOCK_ERROR	
2	HEX	040B	TID_LGLD_RECOVERY_RELEASE_LOCK_ERROR	
2	HEX	040C	TID_LGLD_ADD_SUBPOOL_ERROR	

Len	Type	Value	Name	Description
2	HEX	040D	TID_LGLD_ JOURNALMODELS_ BBLX_EXCEPTION	
2	HEX	040E	TID_LGLD_ JOURNALMODELS_ SIF_EXCEPTION	
2	HEX	040F	TID_LGLD_ BROWSES_BBLX_EXCEPTION	
2	HEX	0410	TID_LGLD_ BROWSES_SIF_EXCEPTION	
2	HEX	0411	TID_LGLD_ JOURNALMODEL_INSTALLED	
2	HEX	0412	TID_LGLD_ JOURNALMODEL_REPLACED	
2	HEX	0413	TID_LGLD_ CATLG_WRITE_ERROR	
2	HEX	0414	TID_LGLD_ CATLG_DELETE_ERROR	
2	HEX	0415	TID_LGLD_ JOURNALMODEL_DISCARDED	

--
-

lgst tracepoints

Len	Type	Value	Name	Description
2	HEX	0501	TID_LGST_ENTRY	
2	HEX	0502	TID_LGST_EXIT	
2	HEX	0503	TID_LGST_RECOVERY	
2	HEX	0504	TID_LGST_ INVALID_FORMAT	
2	HEX	0505	TID_LGST_ INVALID_FUNCTION	
2	HEX	0506	TID_LGST_ UNKNOWN_KE_ERROR_ CODE	
2	HEX	0507	TID_LGST_ GET_EXC_LOCK_ERROR	
2	HEX	0508	TID_LGST_ RELEASE_EXC_LOCK_ ERROR	
2	HEX	0509	TID_LGST_ GET_SHR_LOCK_ERROR	
2	HEX	050A	TID_LGST_ RELEASE_SHR_LOCK_ ERROR	
2	HEX	050B	TID_LGST_ RECOVERY_RELEASE_ LOCK_ERROR	
2	HEX	050C	TID_LGST_ ADD_SUBPOOL_ERROR	
2	HEX	050D	TID_LGST_ STREAMS_BBLX_EXCEPTION	
2	HEX	050E	TID_LGST_ STREAMS_SIF_EXCEPTION	
2	HEX	0510	TID_LGST_ GET_EXC_STREAM_ LOCK_ERROR	
2	HEX	0511	TID_LGST_ RELEASE_EXC_STREAM_ LOCK_ERROR	
2	HEX	0513	TID_LGST_ GET_COND_STREAM_ LOCK_ERROR	
2	HEX	0514	TID_LGST_ STREAM_DEFINED	
2	HEX	0515	TID_LGST_ STREAM_DEFINE_ ERROR	
2	HEX	0516	TID_LGST_ RELEASE_SHR_STREAM_ LOCK_ERROR	
2	HEX	0517	TID_LGST_ STREAM_DEFINE_ INPUT	
2	HEX	0518	TID_LGST_ ENQUEUE_ERROR	
2	HEX	0519	TID_LGST_ DEQUEUE_ERROR	
2	HEX	051A	TID_LGST_ ADD_ENQPOOL_ERROR	
2	HEX	0520	TID_LGST_ ADD_BROWSES_SUBPOOL_ ERROR	
2	HEX	0521	TID_LGST_ BROWSES_BBLX_EXCEPTION	
2	HEX	0522	TID_LGST_ BROWSES_SIF_EXCEPTION	

Len	Type	Value	Name	Description
2	HEX	050F	TID_LGST_ ADD_STREAM_LOCK_ ERROR	
2	HEX	0523	TID_LGST_ REC_RLSE_STREAM_ LOCK_ERROR	
2	HEX	0524	TID_LGST_ CONNECT_ERROR	
2	HEX	0525	TID_LGST_ EXIT_REJECTED_ DEFINE	
2	HEX	0526	TID_LGST_ WAIT_FOR_STREAM_ LOCK	
2	HEX	0527	TID_LGST_ START_WT_BROWSE_ ERROR	
2	HEX	0528	TID_LGST_ GET_NEXT_WT_ERROR	
2	HEX	0529	TID_LGST_ END_WT_BROWSE_ ERROR	
2	HEX	052A	TID_LGST_ GET_EXC_LGUOW_ LOCK_ERROR	
2	HEX	052B	TID_LGST_ RELEASE_EXC_LGUOW_ LOCK_ERROR	
2	HEX	052C	TID_LGST_ REC_RLSE_LGUOW_ LOCK_ERROR	
2	HEX	052D	TID_LGST_ MVS_ENQ_INPUT	
2	HEX	052E	TID_LGST_ MVS_ENQ_OK	
2	HEX	052F	TID_LGST_ MVS_ENQ_FAIL	
2	HEX	0530	TID_LGST_ MVS_DEQ_INPUT	
2	HEX	0531	TID_LGST_ MVS_DEQ_OK	
2	HEX	0532	TID_LGST_ MVS_DEQ_FAIL	

--
-

lgpa tracepoints

Len	Type	Value	Name	Description
2	HEX	0601	TID_LGPA_ENTRY	
2	HEX	0602	TID_LGPA_EXIT	
2	HEX	0603	TID_LGPA_RECOVERY	
2	HEX	0604	TID_LGPA_ INVALID_FORMAT	
2	HEX	0605	TID_LGPA_ INVALID_FUNCTION	

--
-

lgsc tracepoints

Len	Type	Value	Name	Description
2	HEX	0701	TID_LGSC_ENTRY	
2	HEX	0702	TID_LGSC_EXIT	
2	HEX	0703	TID_LGSC_RECOVERY	
2	HEX	0704	TID_LGSC_ INVALID_FORMAT	
2	HEX	0705	TID_LGSC_ INVALID_FUNCTION	
2	HEX	0706	TID_LGSC_ INVALID_PARMS	

LGFL Log of logs failure record

-

The CICS log manager domain will write a record to user journal DFHLGLOG when it detects a write error to any MVS logger log stream. Records will not be written for failed attempts to connect to a log stream.

There will be one record for the stream itself and, if the stream is used as a journal, a record for each CICS journal name that uses the stream.

This record is preceeded by the normal CICS log record header.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	36	LGFL_RECORD	
(0)	UNSIGNED	2	LGFL_DATA_TYPE	Record type
1=Stream Failure 2=Journal Failure				
(2)	CHARACTER	26	LGFL_STREAM_NAME	MVS stream name
(1C)	CHARACTER	8	LGFL_JNL_NAME	Journal name

Constants

Len	Type	Value	Name	Description
2	DECIMAL	1	LGFL_STREAM_ FAIL_REC	
2	DECIMAL	2	LGFL_JNL_FAIL_REC	

LGSF System log format

-

The CICS System Log is a special log where CICS keeps enough data to satisfy the requirements of transaction backout, emergency restart and indoubt resolution. It resides upon the MVS Logger.
The System Log comprises a sequence of contiguous blocks on two physical log streams, the primary and the secondary. Blocks are written to the primary. They may be moved to the secondary at a later point in time so that the tail of the primary can be periodically deleted. This is a performance optimization.

Each block comprises a block header followed by a variable number of CICS records. The format of the block header is defined by the dsect "lgsl_block_header"

Each CICS record comprise a record header followed by caller data normally belonging to CICS Recovery Manager (RM). The record header is defined by the dsect "lgsl_record_header".

The format of the caller data is unknown at the Log Manager functional level. The RM caller data is defined by the Recovery Manager domain.

The following diagram shows the physical layout of a System Log block.

system log
__ first system log block
__ __ block header (lgsl_block_header)
__ __ __ first cics record
__ __ __ __ record header (lgsl_record_header)
__ __ __ __ caller data (RM)
__ __ __ next cics record
__ __ __ ...
__ __ __ last cics record
__ __ __ ...
__ next system log block
__ ...
__ last system log block
__ ...

This copybook defines the block header, record header, and user headers for the System Log.

-

Each block starts with a block header as defined here.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	LGSL_BLOCK_HEADER	
(0)	STRUCTURE	52	*	
	IsA(SYSLOGBLOCKHEADER)			
(0)	CHARACTER	40	SLBH	
(0)	CHARACTER	8	LGBH_GLOBAL_ INFO	
(0)	CHARACTER	4	LGBH_BLOCK_ TYPE	set to '>DFH' to
(0)	CHARACTER	1	LGBH_BT_ ARROW	identify a CICS
(1)	CHARACTER	3	LGBH_BT_DFH	block
(4)	CHARACTER	4	*	
(4)	UNSIGNED	1	LGBH_LOG_ TYPE	general or system log
(5)	CHARACTER	1	LGBH_FLAGS	reserved
(6)	UNSIGNED	2	LGBH_BLOCK_ VER	block format version number
(8)	CHARACTER	24	LGBH_CICS_INFO	
(8)	CHARACTER	8	LGBH_GENERIC_ APPLID	CICS generic applid
(10)	CHARACTER	8	LGBH_START_ GMT	record time (GMT)
(18)	CHARACTER	8	LGBH_START_ LOCAL	record time (LOCAL)
(20)	CHARACTER	8	LGBH_BLOCK_ INFO	
(20)	CHARACTER	8	LGBH_BLOCK_ NUMBER	block sequence number
(28)	CHARACTER		LGBH_DATA	records follow
(28)	CHARACTER	8	SLBH_PREV_ BLOCK_ID	block id prev block
(30)	UNSIGNED	4	SLBH_LAST_ USED_INDEX	index of last record
(34)	CHARACTER		SLBH_DATA	records follow

--
-

Each record starts with a header as defined here, followed by RM data.

The header comprises two parts. The first part is common to all records, and contains a link to the previous record on this logstream. This enables the entire logstream to be sequentially read back on a record basis (during CICS emergency restart). This is known as the 'master chain'.

The second part identifies the different record types. There are four record types, as described below.

- A record written to the primary log as part of a UOW. Contains a link to the previous record in the UOW on the primary.

- A special fork record written to the primary log as part of a UOW. Contains a link to the previous record in the UOW on the primary (the dead tail) and the previous record in the UOW on the secondary.

- A record written to the secondary log as part of a UOW. Contains a link to the previous record in the UOW on the secondary.

- A record written to the primary log by a user and not part of any UOW (unchained).

The UOW links described above enable a UOW to be sequentially read back on a record basis (during dynamic backout). Note that the RM data starts immediately after the chain header finishes, so the RM data starts at a different offset for each different record type.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	68	LGSL_RECORD_ HEADER	
(0)	STRUCTURE	68	* IsA(SYSLOGCOMBINEDRECORD)	
(0)	CHARACTER	16	SLH_PREFIX	initial header
(0)	UNSIGNED	4	SLH_P_REC_LEN	inclusive length of this record
(4)	UNSIGNED	4	SLH_P_HDR_LEN	inclusive length of this header
(8)	CHARACTER	8	SLH_P_STCK	record time (GMT)
(10)	CHARACTER		SLH_P_DATA	start of rest of record
(10)	CHARACTER	16	SLH_MASTER	link to previous
(10)	CHARACTER	16	MASTER_PREV	previous on master chain
(10)	CHARACTER	9	FLAT_BLOCK	block details
(10)	CHARACTER	8	ID_OR_NUMBER	block id or number
(10)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(10)	CHARACTER	8	FLAT_BLOCK_ ID	block id
(18)	CHARACTER	1	BLOCK_ID_ USED	block id used = 'Y', block number used = 'N'
(19)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(1A)	CHARACTER	2	FLAT_RSVD1	reserved
(1C)	UNSIGNED	4	FLAT_INDEX	offset within block
(20)	CHARACTER	36	SLH_REST	record is one of...
(20)	CHARACTER	20	SLH_NORMAL	normal primary
(20)	UNSIGNED	4	REC_TYPE_ NORMAL	normal type (= 1)
(24)	CHARACTER	16	CHAIN_PREV	previous on UOW chain
(24)	CHARACTER	9	FLAT_BLOCK	block details
(24)	CHARACTER	8	ID_OR_NUMBER	block id or number
(24)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(24)	CHARACTER	8	FLAT_BLOCK_ ID	block id
(2C)	CHARACTER	1	BLOCK_ID_ USED	block id used = 'Y', block number used = 'N'
(2D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(2E)	CHARACTER	2	FLAT_RSVD1	reserved
(30)	UNSIGNED	4	FLAT_INDEX	offset within block
(34)	CHARACTER		NORMAL_ RM_START	start of RM data
(20)	CHARACTER	36	SLH_FORK	fork
(20)	UNSIGNED	4	REC_TYPE_FORK	fork type (= 2)
(24)	CHARACTER	16	CHAIN_PREV_ LIVE	previous on UOW chain on secondary
(24)	CHARACTER	9	FLAT_BLOCK	block details
(24)	CHARACTER	8	ID_OR_NUMBER	block id or number
(24)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(24)	CHARACTER	8	FLAT_BLOCK_ ID	block id
(2C)	CHARACTER	1	BLOCK_ID_ USED	block id used = 'Y', block number used = 'N'
(2D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(2E)	CHARACTER	2	FLAT_RSVD1	reserved
(30)	UNSIGNED	4	FLAT_INDEX	offset within block

Offset Hex	Type	Len	Name (Dim)	Description
(34)	CHARACTER	16	CHAIN_PREV_ DEAD	previous on UOW chain on primary
(34)	CHARACTER	9	FLAT_BLOCK	block details
(34)	CHARACTER	8	ID_OR_NUMBER	block id or number
(34)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(34)	CHARACTER	8	FLAT_BLOCK_ ID	block id
(3C)	CHARACTER	1	BLOCK_ID_ USED	block id used = 'Y', block number used = 'N'
(3D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(3E)	CHARACTER	2	FLAT_RSVD1	reserved
(40)	UNSIGNED	4	FLAT_INDEX	offset within block
(44)	CHARACTER		FORK_RM_START	start of RM data
(20)	CHARACTER	20	SLH_SECONDARY	secondary
(20)	UNSIGNED	4	REC_TYPE_SEC	secondary type (= 3)
(24)	CHARACTER	16	CHAIN_PREV_ SEC	previous on UOW chain
(24)	CHARACTER	9	FLAT_BLOCK	block details
(24)	CHARACTER	8	ID_OR_NUMBER	block id or number
(24)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(24)	CHARACTER	8	FLAT_BLOCK_ ID	block id
(2C)	CHARACTER	1	BLOCK_ID_ USED	block id used = 'Y', block number used = 'N'
(2D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(2E)	CHARACTER	2	FLAT_RSVD1	reserved
(30)	UNSIGNED	4	FLAT_INDEX	offset within block
(34)	CHARACTER		SECONDARY_RM_START	start of RM data
(20)	CHARACTER	4	SLH_USER	unchained user
(20)	UNSIGNED	4	REC_TYPE_USER	user type (= 4)
(24)	CHARACTER		USER_RM_START	start of RM data
(20)	CHARACTER	36	SLH_TRIM	unchained trim
(20)	UNSIGNED	4	REC_TYPE_TRIM	trim type (= 5)
(24)	CHARACTER	16	PRIMARY_LOG_HISTORY_POINT_INFO	to trim primary
(24)	CHARACTER	8	PRIMARY_STCK_VALUE	store clock value
(2C)	CHARACTER	8	PRIMARY_ BLOCK_ID	MVS block id
(34)	CHARACTER	16	SECONDARY_LOG_HISTORY_POINT_INFO	to trim secondary
(34)	CHARACTER	8	SECONDARY_STCK_VALUE	store clock value
(3C)	CHARACTER	8	SECONDARY_BLOCK_ID	MVS block id
(44)	CHARACTER		*	
(20)	CHARACTER	20	SLH_NON_MOVED	1ry
(20)	UNSIGNED	4	REC_TYPE_ NORMAL	normal type (= 6)
(24)	CHARACTER	16	CHAIN_PREV	prev on UOW chain
(24)	CHARACTER	9	FLAT_BLOCK	block details
(24)	CHARACTER	8	ID_OR_NUMBER	block id or number
(24)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(24)	CHARACTER	8	FLAT_BLOCK_ ID	block id
(2C)	CHARACTER	1	BLOCK_ID_ USED	block id used = 'Y', block number used = 'N'
(2D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(2E)	CHARACTER	2	FLAT_RSVD1	reserved
(30)	UNSIGNED	4	FLAT_INDEX	offset within block
(34)	CHARACTER		NON_MOVED_RM_START	start of RM data
(44)	CHARACTER		*	

```
--
 -

    The CICS API supports writing directly to the System Log using the
    EXEC CICS WRITE JOURNALNAME command. This takes as input the
    journal type, user data and optional user prefix data. These
  elements are put together in dsect "cl_  user_header" plus some
  extra transaction related data as shown in dsect "sl_ user_header".

  NOTE: "sl_ user_header" followed by "cl_ user_header" form a
    particular case of 'caller data' referred to above. This is the
    only case where caller data is not defined by RM.

    The following diagram shows how the two user headers appear within
    a System Log record.

  system log
  __ ...
    __ system log block
  __ _ _ block header (lgsl_ block_header)
  __ _  _ __ first cics record
  __ __ __ ...
  __ _  _ __ next cics record
  __ _ _ __ __  record header (lgsl_record_header)
  __ _ _ __ __  user header (sl_user_header)
  __ _ _ __ __  user header (cl_user_header)
  __ _ _ __ __  rest of caller data
  __ _  _ __ last cics record
  __ __ __ ...
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	SL_USER_HEADER	
(0)	STRUCTURE IsA(SYSLOGUSER)	16	*	
(0)	CHARACTER	16	SL_UH_TRAN_DATA	
(0)	UNSIGNED	4	SL_UH_TD_ LENGTH	length of this header
(4)	CHARACTER	4	SL_UH_TD_ TASKNO	task number
(8)	CHARACTER	4	SL_UH_TD_ TRANID	tranid
(C)	CHARACTER	4	SL_UH_TD_ TERMID	termid
(10)	CHARACTER		SL_UH_END	general user header follows

```
--
 -

  The CICS API supports writing directly to a user journal (which
  may be a General Log or the System Log) using the EXEC CICS WRITE
  JOURNALNAME command. This takes as input the journal type, user
  data and optional user prefix data. These elements are put
  together as shown in the dsect "cl_user_header".

  NOTE: "cl_user_header" is a particular case of 'caller data'
  referred to above.

  In this case "glrh_rec_compid" will be set to 'UJ'.

  The following diagram shows how a user header appears within a
  General Log record.

  general log
  __ ...
  __ general log block
  __ __ block header (lgbh_block_header)
  __ __ __ first cics record
  __ __ __ ...
  __ __ __ next cics record
  __ __ __ __ record header (glrh_record_header)
  __ __ __ __ user header (cl_user_header)
  __ __ __ __ rest of caller data
  __ __ __ last cics record
  __ __ __ ...

  NOTE: "cl_uh_prefix_length" shows the number of bytes of data that
  is contained in the user prefix. The user prefix data, if present,
  immediately follows this header, which in turn is followed by the
  user data.
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	CL_USER_HEADER	
(0)	STRUCTURE IsA(GENLOGUSER)	12	*	
(0)	UNSIGNED	4	CL_UH_LENGTH	length of structure inclusive of this field
(4)	UNSIGNED	2	CL_UH_JOURNAL_ TYPE	journal type
(6)	CHARACTER	2	CL_UH_RSVD1	reserved
(8)	UNSIGNED	4	CL_UH_PREFIX_ LENGTH	user prefix length
(C)	CHARACTER		CL_UH_END	user prefix data (if any) followed by user data

Constants

Len	Type	Value	Name	Description
2	DECIMAL	1	SLBH_BLOCK_ VERSION_NO	
3	CHARACTER	DFH	SLBH_BLOCK_ TYPE_DFH	
1	CHARACTER	>	SLBH_BLOCK_ TYPE_ARROW	
1	DECIMAL	0	SLBH_LOG_ TYPE_GENERAL	
1	DECIMAL	1	SLBH_LOG_ TYPE_SYSTEM	
4	DECIMAL	1	SLH_P_REC_ TYPE_NORMAL	
4	DECIMAL	2	SLH_P_REC_ TYPE_FORK	
4	DECIMAL	3	SLH_P_REC_ TYPE_SECONDARY	
4	DECIMAL	4	SLH_P_REC_ TYPE_USER	
4	DECIMAL	5	SLH_P_REC_ TYPE_TRIM	
4	DECIMAL	6	SLH_P_REC_ TYPE_NON_MOVED	

LIFO Stack segment table header

CONTROL BLOCK NAME = DFHLIFO
DESCRIPTIVE NAME = CICS (KE) Kernel Lifo control blocks.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS =
MODULE TYPE = Control block definition
EXTERNAL REFERENCES = None.
Segment Entry
Controls the allocation of stack entries within this segment.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	SEGMENT_ENTRY	Segment entry
(0)	CHARACTER	8	SEG_NAME	Eye-catcher SEGENTRY
(8)	ADDRESS	4	SEG_NEXT_FREE	If the segment is free this is the free list pointer
(C)	ADDRESS	4	SEG_CHAIN	If the segment is allocated to a task, this is the segment ownership chain, starting with the current segment
(10)	ADDRESS	4	SEG_START_ OF_SEGMENT	First byte of usable segment storage
(14)	ADDRESS	4	SEG_END_ OF_SEGMENT	Last byte + 1 of this segment
(18)	ADDRESS	4	SEG_CURRENT_ STACK	Current stack in segment
(1C)	BITSTRING	1	SEG_FLAGS	Flags
	1...		SEG_DISPOSABLE	Segment may be freemained *
	.1..		SEG_ACQUIRED_ FROM_SM	Acquired from Stg Mgr
(1D)	BITSTRING	3	*	Reserved
(20)	CHARACTER		SEG_DATA	Start of segment data

Constants

Len	Type	Value	Name	Description
4	DECIMAL	2016	SEGMENT_DATA_LENGTH_24	
4	DECIMAL	28640	SEGMENT_DATA_LENGTH_31	
4	DECIMAL	4064	SEGMENT_DATA_EXTLEN_24	
4	DECIMAL	4064	SEGMENT_DATA_EXTLEN_31	
4	DECIMAL	2147418111	SEGMENT_ADDRESS_LIMIT	
4	DECIMAL	0	SEG_ANYWHERE	
4	DECIMAL	1	SEG_BELOW	

LMCB1 Lock manager domain anchor block

Segment Name = DFHLMCB1
DESCRIPTIVE NAME = CICS Lock Manager Domain Control Blocks 1
Function =
This file contains the data structure
declarations used by the Lock Manager domain.
The file is included by each Lock Manager domain
module.
The data structures are :
ANCHOR - LM Anchor block
LOCK_MANAGEMENT - LM lock management details
LOCK_ELEMENT - LM lock element details
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
LM anchor block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	64	ANCHOR	Anchor Block
(0)	CHARACTER	16	ANC_PREFIX	Anchor prefix area
(0)	HALFWORD	2	ANC_LENGTH	Anchor length
(2)	CHARACTER	1	ANC_ARROW	Arrow eyecatcher
(3)	CHARACTER	3	ANC_DFH	DFH
(6)	CHARACTER	2	ANC_DOMID	Domain id
(8)	CHARACTER	8	ANC_BLOCK_NAME	Control block name
(10)	ADDRESS	4	ANC_QUICKCELL_1_HEAD	-> quickcell 1 head
(14)	ADDRESS	4	ANC_QUICKCELL_2_HEAD	-> quickcell 2 head
(18)	ADDRESS	4	ANC_QUICKCELL_3_HEAD	-> quickcell 3 head
(1C)	UNSIGNED	4	*	Reserved
(20)	CHARACTER	8	ANC_FREECHAIN_1_HEAD	Freechain 1 head
(20)	ADDRESS	4	ANC_FREECHAIN_1_NEXT	-> freechain 1 next
(24)	UNSIGNED	4	ANC_FREECHAIN_1_GUARD	Freechain 1 guard count
(28)	CHARACTER	8	ANC_FREECHAIN_2_HEAD	Freechain 2 head
(28)	ADDRESS	4	ANC_FREECHAIN_2_NEXT	-> freechain 2 next
(2C)	UNSIGNED	4	ANC_FREECHAIN_2_GUARD	Freechain 2 guard count
(30)	CHARACTER	8	ANC_FREECHAIN_3_HEAD	Freechain 3 head
(30)	ADDRESS	4	ANC_FREECHAIN_3_NEXT	-> freechain 3 next
(34)	UNSIGNED	4	ANC_FREECHAIN_3_GUARD	

Offset Hex	Type	Len	Name (Dim)	Description
				Freechain 3 guard count
(38)	UNSIGNED	4	ANC_NUMBER_ OF_LOCKS	
				Number of locks
(3C)	CHARACTER	4	ANC_MAXIMUM_ TASKS	
(3C)	HALFWORD	2	ANC_TASK_LIMIT	mxt task limit
(3E)	HALFWORD	2	ANC_XTRA_LIMIT	overflow allocation

Lock management

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	48	LOCK_MANAGEMENT	Lock Management
(0)	CHARACTER	24	LM_PREFIX	Wait queue prefix area
(0)	HALFWORD	2	LM_LENGTH	Length
(2)	CHARACTER	1	LM_ARROW	Arrow Eyecatcher
(3)	CHARACTER	3	LM_DFH	DFH
(6)	CHARACTER	2	LM_DOMID	Domain id
(8)	CHARACTER	8	LM_BLOCK_NAME	Control block name
(10)	CHARACTER	8	LM_LOCK_NAME	Lock name
(18)	CHARACTER	8	LM_COMP_ AND_SWAP_SECTION	
(18)	FULLWORD	4	LM_CS_OWNER	Owner of x lock
(18)	BITSTRING	1	*	
	1...		LM_CS_MODE_S	'1' shared, '0' excl
	.111 1111		*	Reserved
(19)	BITSTRING	1	*	Reserved
(1A)	HALFWORD	2	LM_CS_COUNT	No. of shared lock users
(1C)	ADDRESS	4	LM_CS_NEXT_PTR	-> to queue of lock waiters
(20)	FULLWORD	4	LM_LOCK_TOKEN	Lock token
(24)	FULLWORD	4	LM_LOCK_REQUESTS	Number of lock requests
(28)	FULLWORD	4	LM_LOCK_SUSPENDS	Number of lock suspends
(2C)	FULLWORD	4	*	Reserved
(30)	CHARACTER		*	

Lock Element

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	20	LOCK_ELEMENT	
(0)	FULLWORD	4	LE_OWNER	Owner of x lock
(0)	BITSTRING	1	*	
	1...		LE_MODE_S	'1' shared, '0' excl
	.111 1111		*	Reserved
(1)	BITSTRING	3	*	Reserved
(4)	ADDRESS	4	LE_NEXT_PTR	-> to queue of lock waiters
(8)	FULLWORD	4	LE_SUSPEND_TOKEN	Suspend_token or 0
(C)	CHARACTER	4	LE_COMP_ AND_SWAP_SECTION	
(C)	BITSTRING	1	*	
	1...		LE_CS_SUSPEND	Compare and swap bit
	.111 1111		*	Reserved
(D)	BITSTRING	3	*	Reserved
(10)	CHARACTER	4	LE_STATUS	
(10)	BITSTRING	1	*	
	1...		LE_DELETED	'1' deleted
	.1..		LE_PURGED	'1' purged
	..11 1111		*	Reserved
(11)	BITSTRING	3	*	Reserved

LMCB2 Lock manager domain quickcell headers

Segment Name = DFHLMCB2
DESCRIPTIVE NAME = CICS Lock Manager Domain Control Blocks 2
Function =
This file contains the data structure
declarations used by the Lock Manager domain.
The data structures are :
QUICKCELL_1 - LM quickcell block descriptor.
QUICKCELL_2 - LM quickcell block descriptor.
QUICKCELL_3 - LM quickcell block descriptor.
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
Quickcell_1
- storage obtained for lock management elements.
A new element is allocated for every add lock.

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(0)	STRUCTURE	24	QUICKCELL_1	
(0)	CHARACTER	24	QUICKCELL_ 1_PREFIX	
(0)	HALFWORD	2	QUICKCELL_ 1_LENGTH	Length
(2)	CHARACTER	1	QUICKCELL_ 1_ARROW	Arrow Eyecatcher
(3)	CHARACTER	3	QUICKCELL_1_DFH	DFH
(6)	CHARACTER	2	QUICKCELL_ 1_DOMID	Domain id
(8)	CHARACTER	8	QUICKCELL_1_BLOCK_NAME	Control block name
(10)	ADDRESS	4	QUICKCELL_ 1_NEXT	-> next
(14)	ADDRESS	4	QUICKCELL_1_LAST_ELEMENT	-> last element

Quickcell_2
- storage obtained for lock queue elements.
A new element is allocated for every wait queue element.

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(0)	STRUCTURE	24	QUICKCELL_2	
(0)	CHARACTER	24	QUICKCELL_ 2_PREFIX	
(0)	HALFWORD	2	QUICKCELL_ 2_LENGTH	Length
(2)	CHARACTER	1	QUICKCELL_ 2_ARROW	Arrow Eyecatcher
(3)	CHARACTER	3	QUICKCELL_2_DFH	DFH
(6)	CHARACTER	2	QUICKCELL_ 2_DOMID	Domain id
(8)	CHARACTER	8	QUICKCELL_2_BLOCK_NAME	Control block name
(10)	ADDRESS	4	QUICKCELL_ 2_NEXT	-> next
(14)	CHARACTER	4	*	Reserved

Quickcell_3
- storage obtained for browse tokens.

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(0)	STRUCTURE	24	QUICKCELL_3	
(0)	CHARACTER	24	QUICKCELL_ 3_PREFIX	
(0)	HALFWORD	2	QUICKCELL_ 3_LENGTH	Length
(2)	CHARACTER	1	QUICKCELL_ 3_ARROW	Arrow Eyecatcher
(3)	CHARACTER	3	QUICKCELL_3_DFH	DFH
(6)	CHARACTER	2	QUICKCELL_ 3_DOMID	Domain id

Offset Hex	Type	Len	Name (Dim)	Description
(8)	CHARACTER	8	QUICKCELL_ 3_BLOCK_NAME	
				Control block name
(10)	ADDRESS	4	QUICKCELL_ 3_NEXT	-> next
(14)	CHARACTER	4	*	Reserved

Quickcell 1 element

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	QUICKCELL_ 1_ELEMENT	
(0)	ADDRESS	4	QUICK_1_ ELEM_NEXT	-> next quickcell element

Quickcell 2 element

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	QUICKCELL_ 2_ELEMENT	
(0)	ADDRESS	4	QUICK_2_ ELEM_NEXT	-> next quickcell element

Quickcell 3 element

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	QUICKCELL_ 3_ELEMENT	
(0)	ADDRESS	4	QUICK_3_ ELEM_NEXT	-> next quickcell element

Freechain 1
- free elements for adding locks

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	FREECHAIN_1	
(0)	ADDRESS	4	FREE_1_NEXT	-> next free element

Freechain 2
- free elements for adding lock elements to the queue

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	FREECHAIN_2	
(0)	ADDRESS	4	FREE_2_NEXT	-> next free element

Freechain 3
- free elements for adding browse tokens

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	FREECHAIN_3	
(0)	ADDRESS	4	FREE_3_NEXT	-> next free element

Constants

Len	Type	Value	Name	Description
4	DECIMAL	18	QUICKMAX_1	Max no. of quickcell elems
4	DECIMAL	18	QUICKMAX_3	Max no. of quickcell elems

L2BL Log manager block class

-

What follows defines the Log Manager Block class.

-

The Block class has instance data and class data.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	104	BLOCK	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	

-

An instance of the Block class consists of...

Offset Hex	Type	Len	Name (Dim)	Description
Declared Data				
(8)	CHAR Protected	92	INSTANCE_ DATA_BLOCK	
(8)	STRUCTURE IsA(L2_EYE_CATCHER) Protected	16	EYE_CATCHER	eye catcher
(8)	UNSIGNED Public	2	L2_EYE_LEN	object length
(A)	UNSIGNED Public	2	L2_EYE_OFFSET	offset of eye-catcher in object
(C)	CHAR Public	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(18)	CHAR Protected	8	BLOCK_NUM	CICS Block Number
(20)	CHAR Protected	8	BLOCK_ID	MVS Block ID
(28)	BITSTRING Protected	1	KNOWN_BY	
	`1... .... Protected`		NUMBER	Block number known
	`.1.. .... Protected`		ID	Block id known
(29)	BITSTRING Protected	1	BTYPE	Flags
	`1... .... Protected`		WRITEABLE	Block used for writing
	`.1.. .... Protected`		READABLE	Block used for reading
	`..1. .... Protected`		UNFLATTENED	Block resulted from unflattening
(2A)	CHAR Protected	2	*	reserved
(2C)	UNSIGNED Protected	4	USE_COUNT	users of this block
(30)	CHAR Protected	8	TIME	time of this block
(38)	STRUCTURE IsA(BLOCKBUFFER) Protected	12	BUFFER	buffer containing data read/written
(38)	ADDRESS Protected	4	START	Start of the buffer
(3C)	SIGNED Protected	4	LEN	Length of the buffer
(40)	ADDRESS Protected	4	CURRENT	Current append point in the buffer
(44)	CHAR Protected	8	JOURNAL_NAME	journal name
(4C)	FIXED Protected	1	SYSLOG	is this part of a system log
(4D)	UNSIGNED Protected	1	STYPE	type of stream
(4E)	CHAR Protected	2	*	reserved
(50)	SIGNED Protected	4	MAX_REC_LEN	maximum record length that could fit in
(54)	CHAR Protected	16	*	reserved

Offset Hex	Type	Len	Name (Dim)	Description

--
-

Declare Block associated types. There are types for BlockContext, BlockBuffer, and ReadCursor. Refer to DFHL2LFC for the definition of Blockid, BlockNumber and FlatBlock.

-

A BlockContext provides context information for a Block object. It is held on Block's behalf by Stream, and is passed to Block on those methods that require context information. Essentially it enables a Block object to know about the other Blocks that have been used by a given Stream.

SHARED DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Public	32	BLOCKCONTEXT	
(0)	CHAR Public	8	CURR_BLOCK_NUM	block number of last block created
(8)	CHAR Public	8	LAST_BLOCK_ID	block id of last block written to MVS
(10)	CHAR Public	8	LAST_BLOCK_TIME	creation time of last block written to MVS
(18)	UNSIGNED Public	1	*	reserved
(19)	UNSIGNED Public	1	*	reserved
(1A)	CHAR Public	6	*	reserved
(20)	CHAR Public		*	

--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	12	BLOCKBUFFER	
(0)	ADDRESS Protected	4	START	Start of the buffer
(4)	SIGNED Protected	4	LEN	Length of the buffer
(8)	ADDRESS Protected	4	CURRENT	Current append point in the buffer
(0)	CHAR Protected	20	READCURSOR	
(0)	ADDRESS Protected	4	BLOCK_PTR	
(4)	ADDRESS Protected	4	HARD_STREAM_PTR	
(8)	CHAR Protected	8	LIMIT_BLOCK_ID	
(10)	STRUCTURE IsA(HSREADTOKEN) Protected	4	HS_READ_TOKEN	

--
-

The class data for the Block class consists of...

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	314	CLASSDATABLOCK	
(0)	STRUCTURE IsA(L2_EYE_CATCHER) Protected	16	CLASS_EYE_ CATCHER	eye catcher
(0)	UNSIGNED Public	2	L2_EYE_LEN	object length
(2)	UNSIGNED Public	2	L2_EYE_OFFSET	offset of eye-catcher in object
(4)	CHAR Public	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(10)	OBJECT IsA(L2OF) Protected	40	OBJECT_FACTORY	object factory for Blocks

-

The instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix 'L2OF' and a suffix which is the name of the object being managed.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHAR Protected	40	INSTANCE_ DATA_BLOCK	L2OF instance data
(10)	CHAR Protected	16	OF_EYE_ CATCHER	eye-catcher
(10)	UNSIGNED Public	2	L2_EYE_LEN	object length
(12)	UNSIGNED Public	2	L2_EYE_ OFFSET	offset of eye-catcher in object
(14)	CHAR Public	12	L2_EYE_ STRING	'>DFHL2xxxxxx'
(20)	CHAR Protected	8	SUBPOOL_NAME	subpool name
(20)	CHAR Protected	4	SUBPOOL_ NAME_PREFIX	subpool name prefix

Offset Hex	Type	Len	Name (Dim)	Description
(24)	CHAR Protected	4	SUBPOOL_ NAME_SUFFIX	subpool name suffix
(28)	CHAR Protected	8	SUBPOOL_TOKEN	subpool token
(30)	CHAR Protected	8	*	
(38)	STRUCTURE IsA(MVSLOGBLOCKHEADER) Protected	40	MVS_BLOCK_ HEADER	
(38)	CHAR Protected	8	LGBH_GLOBAL_ INFO	
(38)	CHAR Protected	4	LGBH_BLOCK_ TYPE	set to '>DFH' to
(38)	CHAR Protected	1	LGBH_BT_ ARROW	identify a CICS
(39)	CHAR Protected	3	LGBH_BT_DFH	block
(3C)	CHAR Protected	4	*	
(3C)	UNSIGNED Protected	1	LGBH_LOG_ TYPE	general or system log
(3D)	CHAR Protected	1	LGBH_FLAGS	reserved
(3E)	UNSIGNED Protected	2	LGBH_BLOCK_ VER	block format version number
(40)	CHAR Protected	24	LGBH_CICS_INFO	
(40)	CHAR Protected	8	LGBH_GENERIC_ APPLID	CICS generic applid
(48)	CHAR Protected	8	LGBH_START_ GMT	record time (GMT)
(50)	CHAR Protected	8	LGBH_START_ LOCAL	record time (LOCAL)
(58)	CHAR Protected	8	LGBH_BLOCK_ INFO	
(58)	CHAR Protected	8	LGBH_BLOCK_ NUMBER	block sequence number
(60)	CHAR Protected		LGBH_DATA	records follow
(60)	STRUCTURE IsA(SMFLOGBLOCKHEADER) Protected	158	SMF_BLOCK_ HEADER	
(60)	CHAR Protected	44	SMF_HEADER	
(60)	UNSIGNED Protected	2	SMFH_LEN	record length
(62)	UNSIGNED Protected	2	SMFH_SEG	segment descriptor
(64)	CHAR Protected	1	SMFH_FLG	operating system indicator (see constant prefixed smfh_flg below)
(65)	CHAR Protected	1	SMFH_RTY	record type (see constant prefixed smfh_rty below)
(66)	CHAR Protected	4	SMFH_TME	time record moved (HHMMSST+)
(6A)	CHAR Protected	4	SMFH_DTE	date record moved (0CYYDDD+)
(6E)	CHAR Protected	4	SMFH_SID	system identification
(72)	CHAR Protected	4	SMFH_SSI	sub-system identification (see constant prefixed smfh_ssi below)
(76)	UNSIGNED Protected	2	SMFH_STY	record subtype (see constant prefixed smfh_sty below)
(78)	UNSIGNED Protected	2	SMFH_TRN	number of triplets in record
(7A)	UNSIGNED Protected	2	SMFH_RSVD1	reserved
(7C)	UNSIGNED Protected	4	SMFH_APS	offset to CICS product section
(80)	UNSIGNED Protected	2	SMFH_LPS	length of CICS product section
(82)	UNSIGNED Protected	2	SMFH_NPS	number of CICS product sections
(84)	UNSIGNED Protected	4	SMFH_ASS	offset to CICS data section
(88)	UNSIGNED Protected	2	SMFH_ASL	length of CICS data section
(8A)	UNSIGNED Protected	2	SMFH_ASN	number of CICS data sections
(8C)	CHAR Protected		*	
(8C)	CHAR Protected	114	SMF_PRODUCT_ SECTION	
(8C)	CHAR Protected	2	SMFPS_VRM	record version format x'0vrm' v = version r = release m = modification (set to &SMF in DFHSYS)
(8E)	CHAR Protected	8	SMFPS_PRN	product name (generic APPLID)
(96)	CHAR Protected	8	SMFPS_SPN	specific APPLID
(9E)	CHAR Protected	2	SMFPS_MFL	record maintenance indicator
(A0)	CHAR Protected	2	SMFPS_RSVD2	reserved
(A2)	CHAR Protected	52	SMFPS_RSVD3	reserved
(D6)	CHAR Protected	8	SMFPS_JNM	journal name
(DE)	CHAR Protected	8	SMFPS_JBN	jobname
(E6)	CHAR Protected	4	SMFPS_RSD	job date
(EA)	CHAR Protected	4	SMFPS_RST	job time
(EE)	CHAR Protected	8	SMFPS_UIF	user identification
(F6)	CHAR Protected	8	SMFPS_PDN	operating system product level
(FE)	CHAR Protected		*	
(FE)	CHAR Protected		SMF_DATA_ SECTION	CICS records
(FE)	CHAR Protected		SMFDS_DATA	records follow
(FE)	STRUCTURE IsA(STARTOFRUNDATA) Protected	20	SOR_DATA	
(FE)	CHAR Protected	20	SOR_CICS_INFO	start-of-run information
(FE)	CHAR Protected	4	SOR_CICS_ RELEASE	CICS version and release

Offset Hex	Type	Len	Name (Dim)	Description
(102)	CHAR Protected	8	SOR_SPECIFIC_ APPLID	CICS specific applid
(10A)	CHAR Protected	8	SOR_CICS_ USERNAME	CICS userid
(112)	CHAR Protected	40	*	Reserved

Constants

Len	Type	Value	Name	Description
4	DECIMAL	1	IO_IN_PROGRESS	
4	DECIMAL	2	LOST_DATA	
4	DECIMAL	3	LOST_ACCESS	
4	DECIMAL	4	DATA_NOT_FOUND	
4	DECIMAL	5	EMPTY_STREAM	
4	DECIMAL	6	END_OF_DATA	

L2BS Log manager browseable stream class

-

The BrowseableStream class declaration contains signatures for the methods, declarations of instance and class data, and implementations of the methods.

-

The BrowseableStream class is declared and is a subclass of the Stream class. Some of Stream's methods are inherited unchanged, others over-ridden and some methods are introduced specific to BrowseableStream.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	728	BROWSEABLESTREAM	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	

-

An instance of Stream class consists of:

- An eyecatcher.

- A double chain link to other streams in the chain of all streams.

- A stream lock which is used to manage concurrent requests made against the stream. Note that a Stream method requiring both the stream lock and the domain lock should acquire the stream lock first to prevent possible deadlock.

- Two block-oriented data structures called StreamBlocks used for managing writes and deferred writes. At any given time one is for the Current block and the other is for the Previous block.

- Pointers to the two StreamBlocks above. One identifies the Current, the other identifies the Previous.

- The ForceToken currently associated with this stream. This is updated on every buffer switch.

- The activity keypoint frequency of the stream, set to zero if activity keypoints do not apply, and an associated count which is used to monitor when activity keypoints are to be triggered.

- Some context data which is owned by the Block class, and is passed to those Block methods that require it.

- The HardStream object that is associated with this stream.

- Whether the stream is an MVS Logger log or an SMF log.

- The logstream name. This is for MVS Logger logs only.

- The journal name. This is a real journal name for SMF logs, or is fabricated from the last qualifier of the logstream name for MVS Logger logs.

- Whether the stream is for a System Log or General Log.

- Some flags indicating progress through the initialization of a Stream object.

- A flag indicating whether the deferred flush mechanism is active for the stream.

- Various statistics for monitoring the number of tasks forced to wait while writing to the stream.

Offset Hex	Type	Len	Name (Dim)	Description
(8)	CHAR Protected	600	STREAM_ INSTANCE_DATA	
(8)	CHAR Protected	16	EYE_CATCHER	an eye-catcher
(8)	UNSIGNED Public	2	L2_EYE_LEN	object length
(A)	UNSIGNED Public	2	L2_EYE_OFFSET	offset of eye-catcher in object
(C)	CHAR Public	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(18)	CHAR Protected	16	STREAM_ CHAIN_LINK	link in global chain
(18)	CHAR Private	4	*	
Declared Data				
(20)	CHAR Protected	8	*	
(20)	ADDRESS Protected	4	PREV	
(24)	ADDRESS Protected	4	NEXT	
(28)	UNSIGNED Protected	4	STREAM_ FORCE_TOKEN	
				Current force token
(2C)	ADDRESS Protected	4	LOCK_TOKEN	stream lock token
(30)	ADDRESS Protected	4	CURRENT	-> Current details
(34)	ADDRESS Protected	4	PREVIOUS	-> Previous details
(38)	CHAR Protected	64	FIRST_BLOCK	Curr or Prev details
(38)	ADDRESS Protected	4	BLOCK_PTR	-> actual Block object
(3C)	UNSIGNED Protected	4	FORCE_TOKEN	force token for block
(40)	ADDRESS Protected	4	NEXT_BLOCK_PTR	-> next Block to be Current
(44)	CHAR Protected	4	BLOCK_OWNER	tran number of nominal owner
(48)	CHAR Protected	40	SUSPEND_QUEUE	chain of suspended tasks

Offset Hex	Type	Len	Name (Dim)	Description
(48)	CHAR Private	4	*	
(50)	CHAR Protected	16	ITER0	
(50)	CHAR Private	4	*	
(58)	CHAR Protected	8	*	
(58)	ADDRESS Protected	4	PREV	
(5C)	ADDRESS Protected	4	NEXT	
(60)	CHAR Protected	16	NODE0	
(60)	CHAR Private	4	*	
(68)	CHAR Protected	8	*	
(68)	ADDRESS Protected	4	PREV	
(6C)	ADDRESS Protected	4	NEXT	
(70)	UNSIGNED Protected	1	STATUS	current status
(71)	CHAR Protected	7	*	
(78)	CHAR Protected	64	SECOND_BLOCK	Curr or Prev details
(78)	ADDRESS Protected	4	BLOCK_PTR	-> actual Block object
(7C)	UNSIGNED Protected	4	FORCE_TOKEN	force token for block
(80)	ADDRESS Protected	4	NEXT_BLOCK_PTR	-> next Block to be Current
(84)	CHAR Protected	4	BLOCK_OWNER	tran number of nominal owner
(88)	CHAR Protected	40	SUSPEND_QUEUE	chain of suspended tasks
(88)	CHAR Private	4	*	
(90)	CHAR Protected	16	ITER0	
(90)	CHAR Private	4	*	
(98)	CHAR Protected	8	*	
(98)	ADDRESS Protected	4	PREV	
(9C)	ADDRESS Protected	4	NEXT	
(A0)	CHAR Protected	16	NODE0	
(A0)	CHAR Private	4	*	
(A8)	CHAR Protected	8	*	
(A8)	ADDRESS Protected	4	PREV	
(AC)	ADDRESS Protected	4	NEXT	
(B0)	UNSIGNED Protected	1	STATUS	current status
(B1)	CHAR Protected	7	*	
(B8)	UNSIGNED Protected	4	AKP_FREQUENCY	activity keypoint frequency
(BC)	SIGNED Protected	4	AKP_COUNT	take keypoint when count reaches zero
(C0)	CHAR Protected	5	BACKTRACK	progress flags
(C0)	UNSIGNED Protected	1	LOCK_ADDED	stream lock added?
(C1)	UNSIGNED Protected	1	CHAINED	on global chain?
(C2)	UNSIGNED Protected	1	CONNECTED	got hard stream?
(C3)	UNSIGNED Protected	1	GOT_BLOCKS	got Curr and Prev?
(C4)	UNSIGNED Protected	1	STATS_OK	gather stats?
(C5)	UNSIGNED Protected	1	LOST_DATA_ WARNING	lost data signalled?
(C6)	UNSIGNED Protected	1	SYSLOG	system log?
(C7)	UNSIGNED Protected	1	TYPE_OF_STREAM	MVS Logger or SMF?
(C8)	CHAR Protected	8	STREAM_JOURNAL	journal name
(D0)	CHAR Protected	32	BLOCK_CONTEXT	block context data owned by Block class
(D0)	CHAR Public	8	CURR_BLOCK_NUM	block number of last block created
(D8)	CHAR Public	8	LAST_BLOCK_ID	block id of last block written to MVS
(E0)	CHAR Public	8	LAST_BLOCK_ TIME	creation time of last block written to MVS
(E8)	UNSIGNED Public	1	*	
(E9)	UNSIGNED Public	1	*	
(EA)	CHAR Public	6	*	
(F0)	CHAR Public		*	
(F0)	CHAR Protected	280	HARD_STREAM	HardStream object
(F0)	CHAR Private	4	*	

Offset Hex	Type	Len	Name (Dim)	Description

-

An instance of HardStream class consists of

- An eyecatcher.

This helps dump navigation.

- A log stream name.

This is the log stream name which denotes the MVS System Logger log stream on connect operation, which returns a log stream token.

- A journal name.

This is the journal name from the log stream name, used as the resource name when a task is suspended.

- A log type.

This is either 'mvs' or 'smf'.

- A connected/disconnected indicator.

When 'connected' the HardStream object is operational, and when 'disconnected' it has been disconnected and it about to be destroyed.

- A System Log indicator.

If 'Y' the log stream forms part of the System Log.

- dasd_ only(y/n)

This flag indicates whether the log stream is of type DASDONLY or CF based.

- structname

If the log stream is CF based, this is the structure name used by the log stream, otherwise this is set to binary 0 (meaning not applicable).

- retention_ period

The log stream retention period is the number in days that the data must be kept before it can be physically deleted by the MVS logger.

- auto_delete

Auto delete flag, if set to yes the MVS logger automatically deletes the data as it matures beyond the retention period, irrespective of any IXGDELET calls. If set to no the data is deleted when it matures beyond the retention period and an IXGDELET call has been issued.

- A maximum block size.

This is a constant, being the maximum block size allowed for the MVS System Logger log stream or MVS SMF log.

- An MVS log stream token.

This is the token that denotes the MVS Logger log stream at its interface. The MVS System Logger returns this value on the connect operation.

- A buffer pointer.

This is the address of the buffer to be written. It is kept here because of the possibility of the need to retry later due to a recoverable error returned from MVS Logger.

- A buffer length.

This is the length of the buffer to be written. It is kept here because of the possibility of the need to retry later due to a recoverable error returned from MVS Logger.

- An ECB.

This is the ECB used when writing to the MVS Logger log stream or MVS SMF log.

- A write answer area.

This is the area where the MVS Logger returns its asynchronous response and diagnostic data.

- A block id.

This is the area where the MVS Logger returns the block id of the block just written.

- A block timestamp.

This is the area where the MVS System Logger returns timestamp of the block just written.

- Warning received indicator.

Set to 'Y' on receipt of a warning exception from the MVS Logger. Reset to 'N' on the first 'ok' response following the warning. Used to limit the number of times a warning message is issued.

- Broken log indicator.

Set to 'Y' on receipt of an unrecoverable error from the MVS Logger. Maintains this state until the log is disconnected. Subsequent calls to a broken log will receive the same response as the original failure, which are kept in the broken response and reason fields.

- Broken response.

- Broken reason.

- SMF response.

This field is the internal response of an SMF write.

- SMF reason.

This field is the internal reason of an SMF write.

- Various statistics.

These are the stats fields that HardStream supports, which are incremented when appropriate and reported/reset on request. For SMF type log streams all stats fields are not used.

- ixg_stck

This is set to the current STCK value just before calling the MVS logger. This is used by the heartbeat task to determine whether it is appropriate to 'touch' the MVS logger.

- ixgwrite_ stck

This is set to the current STCK value just before calling the MVS logger macro IXGWRITE. This is used to evaluate the IXGWRITE latency.

- ixgwrite_ latency

This is set to the time it took to execute the last IXGWRITE call. If the call is made SYNCronously then this is simply the time taken to execute the call and return. If the call is made ASYNCronously then this includes the initial plus the wait period to the posting of the ECB. This is used to cap the LG defer period. This is measured in miliseconds.

Offset Hex	Type	Len	Name (Dim)	Description
(F8)	CHAR Protected	272	INSTANCE_DATA_BLOCK	
(F8)	CHAR Protected	16	EYE_CATCHER	an eye-catcher
(F8)	UNSIGNED Public	2	L2_EYE_LEN	object length
(FA)	UNSIGNED Public	2	L2_EYE_ OFFSET	offset of eye-catcher in object
(FC)	CHAR Public	12	L2_EYE_ STRING	'>DFHL2xxxxxx'
(108)	CHAR Protected	26	MVS_STREAM_ NAME	MVS logstream name
(122)	CHAR Protected	8	JOURNAL_NAME	journal name

Offset Hex	Type	Len	Name (Dim)	Description
(12A)	UNSIGNED Protected	1	LOG_TYPE	log type - MVS or SMF
(12B)	UNSIGNED Protected	1	CONNECTED	connected?
(12C)	UNSIGNED Protected	1	SYSTEM_LOG	CICS system log ind
(12D)	UNSIGNED Protected	1	DASD_ONLY_ FLAG	DASD only flag
(12E)	CHAR Protected	16	STRUCTURE_ NAME	Structure name
(13E)	CHAR Protected	2	*	
(140)	SIGNED Protected	4	RETENTION_ PERIOD	Retention period
(144)	UNSIGNED Protected	1	AUTO_DELETE_ FLAG	Auto delete flag
(145)	CHAR Protected	3	*	
(148)	UNSIGNED Protected	4	MAX_BLOCK_ SIZE	max log block size
(14C)	CHAR Protected	16	MVS_STREAM_ TOKEN	MVS Logger token
(15C)	ADDRESS Protected	4	BUFFER_PTR	write buffer ptr
(160)	UNSIGNED Protected	4	BUFFER_LEN	write buffer length
(164)	UNSIGNED Protected	4	WRITE_ECB	ECB for writing block
(168)	CHAR Protected	40	WRITE_ANSA	ixgwrite answer area
(190)	CHAR Protected	8	CUR_BLOCK_ID	block id
(198)	CHAR Protected	16	CUR_TIMESTAMP	block timestamp
(198)	CHAR Protected	8	CUR_TIME_GMT	GMT time
(1A0)	CHAR Protected	8	CUR_TIME_ LOCAL	local time
(1A8)	UNSIGNED Protected	1	MSL_WARNING_ MSG	warning msg issued
(1A9)	UNSIGNED Protected	1	BROKEN_LOG	log in error flag
(1AA)	CHAR Protected	2	*	
(1AC)	SIGNED Protected	4	BROKEN_RSP	broken response
(1B0)	SIGNED Protected	4	BROKEN_RSN	broken reason
(1B4)	SIGNED Protected	4	SMF_RESPONSE	SMF write response
(1B8)	SIGNED Protected	4	SMF_REASON	SMF write reason
(1BC)	CHAR Protected	29	LOG_STREAM_ STATS	various statistics
(1BC)	SIGNED Protected	4	IXGWRITE_ COUNT	no of writes
(1C0)	BITSTRING Protected	8	IXGWRITE_ BYTES	no of bytes written
(1C8)	SIGNED Protected	4	RETRY_ERRCOUNT	no of retryable errors
(1CC)	SIGNED Protected	4	IXGBROST_ COUNT	no of browse starts
(1D0)	SIGNED Protected	4	IXGBRORD_ COUNT	no of browse reads
(1D4)	SIGNED Protected	4	IXGDELET_ COUNT	no of deletes
(1D8)	UNSIGNED Protected	1	RETRY_ERRCOUNT_ INC_DONE	to ensure stats only incremented once
(1D9)	CHAR Protected	7	*	
(1E0)	CHAR Protected	8	IXG_STCK	Timestamp of last call
(1E8)	CHAR Protected	8	IXGWRITE_STCK	IXGWRITE timestamp
(1F0)	UNSIGNED Protected	4	IXGWRITE_ LATENCY	IXGWRITE latency
(1F4)	CHAR Protected	20	*	
(208)	CHAR Protected	26	LOGSTREAM_NAME	logstream name
(222)	CHAR Protected	2	*	
(224)	CHAR Protected	28	LOGSTREAM_STATS	statistics
(224)	SIGNED Protected	4	FORCE_WAITS_CU	current, peak and
(228)	SIGNED Protected	4	FORCE_WAITS_PK	total waiters for
(22C)	SIGNED Protected	4	FORCE_WAITS_TO	Current buffer force
(230)	SIGNED Protected	4	BUF_FULL_WAITS	total waiters for Previous buffer write
(234)	SIGNED Protected	4	BUF_APPENDS	No of buffer appends
(238)	CHAR Protected	8	*	
(240)	UNSIGNED Protected	4	*	

Offset Hex	Type	Len	Name (Dim)	Description
(240)	UNSIGNED Protected	1	DEFER_FORCE_ FLAG	active flag. 31 bits resvd.
(244)	CHAR Protected	4	*	
(248)	CHAR Protected	24	LOGSTREAM_ OPT_FIELDS	Wait optimiser
(248)	CHAR Protected	6	*	
(24E)	CHAR Protected	8	INTERVAL_START	STCK of start
(24E)	UNSIGNED Protected	2	START_HIGH	High order hword
(250)	UNSIGNED Protected	4	START_TIME	16 microsecond units
(254)	CHAR Protected	2	*	
(256)	CHAR Protected	2	*	
(258)	SIGNED Protected	4	LAST_FORCE_ TASK	Last forcing tsk
(25C)	SIGNED Protected	4	AVERAGE_GAP	Average gap
(260)	CHAR Protected		*	

-

In addition to the instance data inherited from the Stream class, instances of the BrowseableStream class consist of:

- an eyecatcher,

- a double chain link to other browseable streams in the chain of all browseable streams,

- a record token pointing to the head of the master chain of records,

- a record token pointing to the next record to be read as part of a master chain browse of records on this browseable stream.

- some flags indicating progress through the initialisation of a browseable stream object,

- some flags set aside for general use,

- some space reserved for future use.

(260)	CHAR Protected	120	BROWSEABLE_ STREAM_ INSTANCE_DATA	
(260)	STRUCTURE IsA(L2_EYE_CATCHER) Protected	16	BSID_EYE_ CATCHER	eye-catcher
(260)	UNSIGNED Public	2	L2_EYE_LEN	object length
(262)	UNSIGNED Public	2	L2_EYE_OFFSET	offset of eye-catcher in object
(264)	CHAR Public	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(270)	OBJECT IsA(HOP_DCHAINNODE) Protected	16	BSID_CHAIN_LINK	link in chain of browseable streams
(270)	CHAR Private	4	*	
(278)	CHAR Protected	8	*	
(278)	ADDRESS Protected	4	PREV	
(27C)	ADDRESS Protected	4	NEXT	
(280)	OBJECT IsA(RECORDTOKEN) Protected	24	BSID_CHAIN_HEAD	head of master chain of records
(280)	CHAR Private	4	*	

-

An instance of the RecordToken class consists of a pointer to the associated Block object, and an index which is the offset of the record within that block. Note that the largest size block that MVS allows is 64K bytes.

A null RecordToken has no underlying Block and so has a null pointer and an index of zero.

(288)	CHAR Protected	10	INSTANCE_ DATA_BLOCK	
(288)	ADDRESS Protected	4	BLOCK_PTR	pointer to Block object
(28C)	UNSIGNED Protected	4	INDEX	offset within block
(290)	CHAR Protected	2	*	

Offset Hex	Type	Len	Name (Dim)	Description
(298)	OBJECT IsA(RECORDTOKEN) Protected	24	BSID_NEXT_ RTOKEN	next record token in chain browse
(298)	CHAR Private	4	*	
(2A0)	CHAR Protected	10	INSTANCE_ DATA_BLOCK	
(2A0)	ADDRESS Protected	4	BLOCK_PTR	pointer to Block object
(2A4)	UNSIGNED Protected	4	INDEX	offset within block
(2A8)	CHAR Protected	2	*	
(2B0)	CHAR Protected	4	BSID_BACKTRACK	progress flags
(2B0)	FIXED Protected	1	BSID_CHAINED	on master chain?
(2B1)	CHAR Protected	3	*	reserved
(2B4)	CHAR Protected	4	BSID_FLAGS	general flags
(2B4)	FIXED Protected	1	BSID_BROWSE_ IN_PROGRESS	master chain browse in progress?
(2B5)	FIXED Protected	1	BSID_EMPTY_ STREAM	empty at startup?
(2B6)	CHAR Protected	2	*	reserved
(2B8)	CHAR Protected	32	*	reserved
(2D8)	CHAR Protected		*	round to double word

SHARED DATA

Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	ADDRESS Public	4	BRLOGSTREAMTOKEN	

--
-

The BrowseableStream class data consists of:

- an eyecatcher,

- the anchor of a doubly-linked list of all browseable streams,

- an object factory instance used to allocate browseable stream instances,

- some space reserved for future use.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	128	BROWSEABLE_ STREAM_CLASS_DATA	
(0)	STRUCTURE IsA(L2_EYE_CATCHER) Protected	16	BSCD_EYE_ CATCHER	eye-catcher
(0)	UNSIGNED Public	2	L2_EYE_LEN	object length
(2)	UNSIGNED Public	2	L2_EYE_OFFSET	offset of eye-catcher in object
(4)	CHAR Public	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(10)	OBJECT IsA(HOP_DCHAIN) Protected	40	BSCD_CHAIN	anchor for chain of browseable streams

Inherited Data

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHAR Private	4	*	
(18)	CHAR Protected	16	ITER0	
(18)	CHAR Private	4	*	
(20)	CHAR Protected	8	*	
(20)	ADDRESS Protected	4	PREV	
(24)	ADDRESS Protected	4	NEXT	
(28)	CHAR Protected	16	NODE0	
(28)	CHAR Private	4	*	
(30)	CHAR Protected	8	*	
(30)	ADDRESS Protected	4	PREV	
(34)	ADDRESS Protected	4	NEXT	
(38)	OBJECT IsA(L2OF) Protected	40	BSCD_FACTORY	browseable stream factory instance

-

The instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix 'L2OF' and a suffix which is the name of the object being managed.

Offset Hex	Type	Len	Name (Dim)	Description
(38)	CHAR Protected	40	INSTANCE_ DATA_BLOCK	

Offset Hex	Type	Len	Name (Dim)	Description
				L2OF instance data
(38)	CHAR Protected	16	OF_EYE_ CATCHER	eye-catcher
(38)	UNSIGNED Public	2	L2_EYE_LEN	object length
(3A)	UNSIGNED Public	2	L2_EYE_ OFFSET	offset of eye-catcher in object
(3C)	CHAR Public	12	L2_EYE_ STRING	'>DFHL2xxxxxx'
(48)	CHAR Protected	8	SUBPOOL_NAME	subpool name
(48)	CHAR Protected	4	SUBPOOL_ NAME_PREFIX	subpool name prefix
(4C)	CHAR Protected	4	SUBPOOL_ NAME_SUFFIX	subpool name suffix
(50)	CHAR Protected	8	SUBPOOL_TOKEN	subpool token
(58)	CHAR Protected	8	*	
(60)	CHAR Protected	32	*	reserved
(80)	CHAR Protected		*	round to double word

Constants

Len	Type	Value	Name	Description
4	DECIMAL	102	NO_BROWSE_ IN_PROGRESS	
4	DECIMAL	101	BROWSE_ALREADY_ IN_PROGRESS	

L2CH Log manager chain class

-

The L2CH Class declaration contains the signatures for the methods, the declaration of the instance and class data, and the implementations of the internal, inlineable methods.

The copybook protects itself against multiple inclusion.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	248	CHAIN	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	

-

An instance of Chain class consists of:

- an eyecatcher,
- a link allowing the instance to be collected into the global list of chains,
- a link allowing the instance to be placed on a free list of chains,
- a record token object referring to the last record written to the chain (the 'head' of the chain).
- the log stream token of the primary system log stream,
- primary and secondary system log stream history points,
- a lock manager lock to enable access to the chain to be serialised,
- flags: whether or not the instance is on the free chain, whether or not the chain is active (an inactive chain exists just to assist the backwards scan of the log during system restart), whether or not the primary log is a dummy, and whether or not a chain browse is processing the secondary log,
- a record token referring to the next record to be read by a chain browse,
- read tokens for primary and secondary log stream browses which are used to browse the chain,
- reserved space to be used for APAR fixes etc. which want to avoid causing large numbers of recompilations.

NOTE: All the instances of chain are kept on the global list of chains. Those that are on the free chain are flagged so that their 'allocated' bit is zero. This avoids the overhead of adding and removing chains from the global list during typical create() and destroy() method calls. A consequence is that unallocated chains must be skipped in all browses of the global list.

Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(8)	CHAR Protected	236	INSTANCE_ DATA_BLOCK	
(8)	STRUCTURE IsA(L2_EYE_CATCHER) Protected	16	EYE_CATCHER	An eye-catcher
(8)	UNSIGNED Public	2	L2_EYE_LEN	object length
(A)	UNSIGNED Public	2	L2_EYE_OFFSET	offset of eye-catcher in object
(C)	CHAR Public	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(18)	OBJECT IsA(HOP_DCHAINNODE) Protected	16	CHAIN_LIST_LINK	Link in global list
(18)	CHAR Private	4	*	
(20)	CHAR Protected	8	*	
(20)	ADDRESS Protected	4	PREV	
(24)	ADDRESS Protected	4	NEXT	
(28)	ADDRESS Protected	4	CHAIN_FREE_ LIST_LINK	Link in free list
(2C)	ADDRESS Protected	4	LOCK_TOKEN	Chain lock token
(30)	ADDRESS Protected	4	PRIMARY_LOG	Primary log stream
(34)	ADDRESS Protected	4	USER_TOKEN	User Token
(38)	SIGNED Protected	4	CURRENT_STREAM	Current stream being read
(3C)	BITSTRING Protected	1	FLAGS	Flags
	1... .... Protected		ALLOCATED	not on free chain
	.1.. .... Protected		ACTIVE	Chain active
	..1. .... Protected		DUMMY_PRIMARY	Primary log is dummy
	...1 .... Protected		SEC_BROWSE	Browsing secondary log
	.... 1... Protected		MOVE_IN_ PROGRESS	Records being copied to secondary stream
	.... .111 Protected		*	Reserved
(3D)	BITSTRING Protected	1	RECOVERY_FLAGS	Flags for recovery
	1... .... Protected		RESTORED	Chain has been restored

Offset Hex	Type	Len	Name (Dim)	Description
	.1.. .... Protected		RECOVERED	Chain recovered from log
	..1. .... Protected		DESTROY	Chain must be destroyed
	...1 .... Protected		IN_DEAD_TAIL	browse_all might find dead tails records @PBA
	.... 1111 Protected		*	Reserved
(3E)	CHAR Protected	2	*	Reserved
(40)	CHAR Protected	80	STREAM_ RESOURCES (2)	One struct for each stream
(40)	OBJECT IsA(RECORDTOKEN) Protected	24	HEAD	Head of chain on stream
(40)	CHAR Private	4	*	

-

An instance of the RecordToken class consists of a pointer to the associated Block object, and an index which is the offset of the record within that block. Note that the largest size block that MVS allows is 64K bytes.

A null RecordToken has no underlying Block and so has a null pointer and an index of zero.

Offset Hex	Type	Len	Name (Dim)	Description
(48)	CHAR Protected	10	INSTANCE_ DATA_BLOCK	
(48)	ADDRESS Protected	4	BLOCK_PTR	pointer to Block object
(4C)	UNSIGNED Protected	4	INDEX	offset within block
(50)	CHAR Protected	2	*	
(58)	OBJECT IsA(HISTORYPOINT) Protected	24	HP	History Point

-

An instance of the HistoryPoint class consists of a store clock value, a block id, and a history point type.

There are three different history point types:

- Ultimate past. This is the earliest possible history point, and has a low values store clock and a null block id.

- Normal. This is a history point strictly between ultimate past and ultimate future, and has a real store clock and a real block id.

- Ultimate future. This is the latest possible history point, and has a high values store clock and a null block id.

Offset Hex	Type	Len	Name (Dim)	Description
(58)	CHAR Protected	24	INSTANCE_ DATA_BLOCK	
(58)	CHAR Protected	8	STCK_VALUE	store clock value
(60)	CHAR Protected	8	BLOCK_ID	block id
(68)	UNSIGNED Protected	1	TYPE	history point type
(69)	CHAR Protected	7	*	
(70)	ADDRESS Protected	4	BROWSE	stream browse token
(74)	SIGNED Protected	4	RECORD_COUNT	Number of records
(78)	OBJECT IsA(RECORDTOKEN) Protected	24	NEXT_IN_BROWSE	Next record to browse
(78)	CHAR Private	4	*	
(80)	CHAR Protected	10	INSTANCE_ DATA_BLOCK	
(80)	ADDRESS Protected	4	BLOCK_PTR	pointer to Block object
(84)	UNSIGNED Protected	4	INDEX	offset within block
(88)	CHAR Protected	2	*	
(E0)	CHAR Protected	20	*	Reserved

SHARED DATA

Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	40	RECORDSTACKELEMENT	
(0)	OBJECT IsA(HOP_DCHAINNODE) Protected	16	LINK	

Inherited Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Private	4	*	
(8)	CHAR Protected	8	*	
(8)	ADDRESS Protected	4	PREV	
(C)	ADDRESS Protected	4	NEXT	

Offset Hex	Type	Len	Name (Dim)	Description
(10)	OBJECT IsA(RECORDTOKEN) Protected	24	RECORD_TOKEN	
(10)	CHAR Private	4	*	
(18)	CHAR Protected	10	INSTANCE_ DATA_BLOCK	
(18)	ADDRESS Protected	4	BLOCK_PTR	pointer to Block object
(1C)	UNSIGNED Protected	4	INDEX	offset within block
(20)	CHAR Protected	2	*	

--
-

The class data of a class is its own anchor block which is shared between all instances of the class.

The Chain class data consists of:

- an eyecatcher,

- the anchor of a doubly-linked list of all the chains in use,

- an object factory instance used to allocate chain instances,

- a list of free chain instances (each with associated resources e.g. a lock manager lock),

- information relating to browse all such as the status of browse all, an iterator used to browse the list of chains, and read tokens for the primary and secondary log browses,

- reserved space to be used for APAR fixes etc. which want to avoid causing large numbers of recompilations.

(0)	CHAR Protected	256	CLASSDATABLOCK	
(0)	STRUCTURE IsA(L2_EYE_CATCHER) Protected	16	CLASS_EYE_ CATCHER	An eye-catcher
(0)	UNSIGNED Public	2	L2_EYE_LEN	object length
(2)	UNSIGNED Public	2	L2_EYE_OFFSET	offset of eye-catcher in object
(4)	CHAR Public	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(10)	CHAR Protected	84	CHAIN_MANAGMENT	
(10)	OBJECT IsA(HOP_DCHAIN) Protected	40	GLOBAL_ CHAIN_LIST	
				All chains
(10)	CHAR Private	4	*	
(18)	CHAR Protected	16	ITER0	
(18)	CHAR Private	4	*	
(20)	CHAR Protected	8	*	
(20)	ADDRESS Protected	4	PREV	
(24)	ADDRESS Protected	4	NEXT	
(28)	CHAR Protected	16	NODE0	
(28)	CHAR Private	4	*	
(30)	CHAR Protected	8	*	
(30)	ADDRESS Protected	4	PREV	
(34)	ADDRESS Protected	4	NEXT	
(38)	OBJECT IsA(L2OF) Protected	40	CHAIN_FACTORY	Chain factory

-

The instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix 'L2OF' and a suffix which is the name of the object being managed.

(38)	CHAR Protected	40	INSTANCE_ DATA_BLOCK	
				L2OF instance data
(38)	CHAR Protected	16	OF_EYE_ CATCHER	eye-catcher
(38)	UNSIGNED Public	2	L2_EYE_LEN	object length
(3A)	UNSIGNED Public	2	L2_EYE_ OFFSET	offset of eye-catcher in object
(3C)	CHAR Public	12	L2_EYE_ STRING	'>DFHL2xxxxxx'
(48)	CHAR Protected	8	SUBPOOL_NAME	subpool name

Offset Hex	Type	Len	Name (Dim)	Description
(48)	CHAR Protected	4	SUBPOOL_ NAME_PREFIX	subpool name prefix
(4C)	CHAR Protected	4	SUBPOOL_ NAME_SUFFIX	subpool name suffix
(50)	CHAR Protected	8	SUBPOOL_ TOKEN	subpool token
(58)	CHAR Protected	8	*	
(60)	ADDRESS Protected	4	CHAIN_FREE_ LIST	Head of free list
(64)	BITSTRING Protected	1	CLASS_FLAGS	Flags
	1... .... Protected		BROWSE_ALL	Browse all mode
	.1.. .... Protected		CLASS_SEC_ BROWSE	
(65)	CHAR Protected	3	*	Reserved
(68)	CHAR Protected	24	CHAINS_ BROWSE_RESOURCES	
(68)	STRUCTURE IsA(ITERATOR) Protected	24	CHAINS_ITER	Chains iterator
(68)	CHAR Public	16	ITERNODE	
(68)	CHAR Private	4	*	
(70)	CHAR Protected	8	*	
(70)	ADDRESS Protected	4	PREV	
(74)	ADDRESS Protected	4	NEXT	
(78)	ADDRESS Public	4	CURRNODE	
(7C)	ADDRESS Public	4	CHAIN_PTR	
(80)	CHAR Protected	16	CLASS_BROWSE_ RESOURCES	
(80)	ADDRESS Protected	4	CLASS_PRIMARY_ BROWSE	Primary stream browse
(84)	ADDRESS Protected	4	CLASS_SECONDARY_ BROWSE	Secondary stream browse
(88)	ADDRESS Protected	4	CURRENT_ CHAIN_PTR	
(8C)	CHAR Protected	4	*	Reserved
(90)	CHAR Protected	56	HISTORY_ POINT_INFO	
(90)	OBJECT IsA(HISTORYPOINT) Protected	24	CURRENT_HP (2)	Current History Point
(90)	CHAR Protected	24	INSTANCE_ DATA_BLOCK	
(90)	CHAR Protected	8	STCK_VALUE	store clock value
(98)	CHAR Protected	8	BLOCK_ID	block id
(A0)	UNSIGNED Protected	1	TYPE	history point type
(A1)	CHAR Protected	7	*	
(C0)	FIXED Protected	1	HISTORY_ POINTS_RESTORED	Have HPs been restored yet during a restart?
(C1)	FIXED Protected	1	HP_TRIMMED_TO (2)	Has HP been used to trim the log to?
(C3)	CHAR Protected	5	*	Reserved
(C8)	CHAR Protected	16	TIME_OF_ LAST_MOVE	Info on last move
(C8)	CHAR Protected	8	START	Time started
(D0)	CHAR Protected	8	FINISH	Time finished
(D8)	CHAR Protected	32	*	Reserved
(F8)	SIGNED Protected	4	COUNT	number of records read
(FC)	UNSIGNED Protected	4	AKP_FREQUENCY	

Constants

Len	Type	Value	Name	Description
4	CHARACTER	ALG8	L2CH_LOCK_ ERROR_CODE	
4	CHARACTER	ALG9	L2CH_UNLOCK_ ERROR_CODE	
	Following raised in development environment only			
4	CHARACTER	ALG	L2CH_WRONG_ TCB_ERROR_CODE	
4	DECIMAL	1	BUFFER_FULL	
4	DECIMAL	2	AKP_KICK_OFF	
4	DECIMAL	4	BUFFER_LENGTH_ ERROR	
4	DECIMAL	5	BROWSE_ILLOGIC	
4	DECIMAL	3	END_OF_DATA	
4	DECIMAL	6	DUMMY_SECONDARY_ STREAM	

L2DM Log manager l2dm class

-

The L2DM Class declaration contains the signatures for the methods and the declaration of the instance data. The instance data structure is the L2 portion of the LG Domain anchor block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	1024	L2DM	

--
-

This structure is the global data for the L2 portion of LG Domain. It occupies the second 1K bytes of the overall LG anchor block (LGA, mapped by copybook DFHLGANC).

INSTANCE DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	1024	INSTANCE_ DATA_BLOCK	
(0)	STRUCTURE IsA(L2_EYE_CATCHER) Protected	16	L2DM_EYE_ CATCHER	Eyecatcher
(0)	UNSIGNED Public	2	L2_EYE_LEN	object length
(2)	UNSIGNED Public	2	L2_EYE_OFFSET	offset of eye-catcher in object
(4)	CHAR Public	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(10)	UNSIGNED Public	1	L2DM_STATE	State
(11)	CHAR Protected	3	*	Reserved
(14)	CHAR Protected	8	L2DM_SUBPOOL	Subpool Token
(1C)	ADDRESS Protected	4	L2DM_LOCK_TOKEN	Domain Lock Token
(20)	OBJECT IsA(RMCLM) Protected	144	L2DM_CLASS_ MANAGER	Class Manager
(20)	CHAR Protected	144	INSTANCE_ DATA_BLOCK	
(20)	CHAR Protected	4	NAME (12)	class name
(50)	ADDRESS Protected	4	INITIALISER (12)	class initialising proc
(80)	ADDRESS Protected	4	DATA (12)	class data address
(B0)	ADDRESS Protected	4	HEARTBEAT_ SUSPEND_TOKEN	Suspend token @P2A
(B4)	CHAR Protected	8	*	reserved

--
lock status type

SHARED DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	BITSTRING	1	L2DM_LOCK_STATUS	
	Public			
	1... .... Protected		HELD	
	.111 1111 Protected		*	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	12	RMCLM_MAX_CLASS	Capacity of the class mgr
4	DECIMAL	6	L2DM_NUM_CLASSES	Number of L2 classes
L2 Classes identified by constant				
4	DECIMAL	1	L2VP_CLASSID	
4	DECIMAL	2	L2BL_CLASSID	
4	DECIMAL	3	L2SR_CLASSID	
4	DECIMAL	4	L2BS_CLASSID	
4	DECIMAL	5	L2SL_CLASSID	
4	DECIMAL	6	L2CH_CLASSID	
4	DECIMAL	0	L2DM_LOCK_FREE	
4	DECIMAL	128	L2DM_LOCK_HELD	
lock error codes				
4	CHARACTER	AL2A	L2DM_LOCK_ ERROR_CODE	
4	CHARACTER	AL2B	L2DM_UNLOCK_ ERROR_CODE	
persistent name and persistent type				
8	CHARACTER	DFHL2DM	L2DM_PTYPE	
16	CHARACTER	DFHL2DM_ANCHOR	L2DM_PNAME	
states				
4	DECIMAL	1	L2DM_INITIALISING	
4	DECIMAL	2	L2DM_INITIALISED	
4	DECIMAL	3	L2DM_QUIESCING	
4	DECIMAL	4	L2DM_QUIESCED	
4	DECIMAL	5	L2DM_TERMINATING	
4	DECIMAL	6	L2DM_TERMINATED	
4	DECIMAL	1	RMCLM_OK	
1	BIT	00000000	LMLM_LOCK_FREE	
1	BIT	10000000	LMLM_LOCK_HELD	

L2HP Log manager history point class

-

What follows defines the Log Manager HistoryPoint class.

History points provide a means of remembering the age of records written to logs. They are used by the System Log class and the Chain class, so are only of relevance to the system log.

The history point of a log record consists of the store clock value that was stored in the record when it was written to the buffer together with a block id, where the block id is not later than the block containing the record.

The history points of a chain are the history points of the oldest records on the primary and secondary log streams belonging to the live part of the chain. If there is no oldest record on either log stream the corresponding history point is in the 'ultimate future' (the latest possible history point).

If the oldest block id is unknown then the history point is in the 'ultimate past' (the oldest possible history point). This occurs, for example, during browse all when the first record of the chain has not yet been browsed, or on a very early write to a log stream after a cold start.

The current history point of a log stream is the history point of the most recently written record on that log stream. If the most recently written record is unknown, then the history point is in the ultimate past. An empty log stream is an example of this.

-

The HistoryPoint class has instance data but no class data.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	24	HISTORYPOINT	

-

An instance of the HistoryPoint class consists of a store clock value, a block id, and a history point type.

There are three different history point types:

- Ultimate past. This is the earliest possible history point, and has a low values store clock and a null block id.

- Normal. This is a history point strictly between ultimate past and ultimate future, and has a real store clock and a real block id.

- Ultimate future. This is the latest possible history point, and has a high values store clock and a null block id.

INSTANCE DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	24	INSTANCE_ DATA_BLOCK	
(0)	CHAR Protected	8	STCK_VALUE	store clock value
(8)	CHAR Protected	8	BLOCK_ID	block id
(10)	FIXED Protected	1	TYPE	history point type
(11)	CHAR Protected	7	*	reserved

--
-

Declare HistoryPoint associated types. There is a type for history point type.

SHARED DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	FIXED Public	1	HPTYPE	

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	HP_ULTIMATE_PAST	
1	DECIMAL	2	HP_NORMAL	
1	DECIMAL	3	HP_ULTIMATE_FUTURE	
8	CHARACTER		ULT_PAST_STCK	
8	CHARACTER		ULT_FUTURE_STCK	

L2HS Log manager hard stream class

-

The HardStream Class declaration contains the signatures for the methods, the declaration of the instance data, and the implementations of the internal methods.

This class provides the following operations, all of which operate on a single object of the HardStream class;-

- Connect

Connect to the MVS Logger or SMF logstream and initialize the HardStream object.

- Disconnect

Disconnect from the logstream and destroy the HardStream object.

- Delete_all

Delete all blocks of data from the logstream (MVS Logger only).

- Delete_history

Delete all blocks of data from the logstream that are strictly older than a specified block (MVS Logger only).

- Get_block_size

Returns the maximum block size allowed for the logstream.

- Get_current_block

Returns the block id and block of the youngest block on the logstream (MVS Logger only).

- Start_read

Start a browse in order to read blocks back from the logstream (MVS Logger only).

- Read_block

Read a specified block from the logstream (MVS Logger only).

- End_read

End a browse.

- Start_write

Write a block of data to the logstream without waiting for the result. A subsequent wait_write operation is used to obtain the result.

- Wait_write

Obtain the result of a previously issued write of a block of data, waiting for the write to complete if necessary.

- Collect_statistics

Return statistics data for the logstream (MVS Logger only).

- Reset_statistics

Reset statistics data for the logstream (MVS Logger only).

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	280	HARDSTREAM	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	

-

An instance of HardStream class consists of

- An eyecatcher.

This helps dump navigation.

- A log stream name.

This is the log stream name which denotes the MVS System Logger log stream on connect operation, which returns a log stream token.

- A journal name.

This is the journal name from the log stream name, used as the resource name when a task is suspended.

- A log type.

This is either 'mvs' or 'smf'.

- A connected/disconnected indicator.

When 'connected' the HardStream object is operational, and when 'disconnected' it has been disconnected and it about to be destroyed.

- A System Log indicator.

If 'Y' the log stream forms part of the System Log.

- dasd_ only(y/n)

This flag indicates whether the log stream is of type DASDONLY or CF based.

- structname

If the log stream is CF based, this is the structure name used by the log stream, otherwise this is set to binary 0 (meaning not applicable).

- retention_ period

The log stream retention period is the number in days that the data must be kept before it can be physically deleted by the MVS logger.

- auto_delete

Auto delete flag, if set to yes the MVS logger automatically deletes the data as it matures beyond the retention period, irrespective of any IXGDELET calls. If set to no the data is deleted when it matures beyond the retention period and an IXGDELET call has been issued.

- A maximum block size.

This is a constant, being the maximum block size allowed for the MVS System Logger log stream or MVS SMF log.

- An MVS log stream token.

This is the token that denotes the MVS Logger log stream at its interface. The MVS System Logger returns this value on the connect operation.

- A buffer pointer.

This is the address of the buffer to be written. It is kept here because of the possibility of the need to retry later due to a recoverable error returned from MVS Logger.

- A buffer length.

This is the length of the buffer to be written. It is kept here because of the possibility of the need to retry later due to a recoverable error returned from MVS Logger.

Offset Hex	Type	Len	Name (Dim)	Description

- An ECB.

This is the ECB used when writing to the MVS Logger log stream or MVS SMF log.

- A write answer area.

This is the area where the MVS Logger returns its asynchronous response and diagnostic data.

- A block id.

This is the area where the MVS Logger returns the block id of the block just written.

- A block timestamp.

This is the area where the MVS System Logger returns timestamp of the block just written.

- Warning received indicator.

Set to 'Y' on receipt of a warning exception from the MVS Logger. Reset to 'N' on the first 'ok' response following the warning. Used to limit the number of times a warning message is issued.

- Broken log indicator.

Set to 'Y' on receipt of an unrecoverable error from the MVS Logger. Maintains this state until the log is disconnected. Subsequent calls to a broken log will receive the same response as the original failure, which are kept in the broken response and reason fields.

- Broken response.

- Broken reason.

- SMF response.

This field is the internal response of an SMF write.

- SMF reason.

This field is the internal reason of an SMF write.

- Various statistics.

These are the stats fields that HardStream supports, which are incremented when appropriate and reported/reset on request. For SMF type log streams all stats fields are not used.

- ixg_stck

This is set to the current STCK value just before calling the MVS logger. This is used by the heartbeat task to determine whether it is appropriate to 'touch' the MVS logger.

- ixgwrite_ stck

This is set to the current STCK value just before calling the MVS logger macro IXGWRITE. This is used to evaluate the IXGWRITE latency.

- ixgwrite_ latency

This is set to the time it took to execute the last IXGWRITE call. If the call is made SYNCronously then this is simply the time taken to execute the call and return. If the call is made ASYNCronously then this includes the initial plus the wait period to the posting of the ECB. This is used to cap the LG defer period. This is measured in miliseconds.

Declared Data

(8)	CHAR Protected	272	INSTANCE_ DATA_BLOCK	
(8)	STRUCTURE IsA(L2_EYE_CATCHER) Protected	16	EYE_CATCHER	an eye-catcher
(8)	UNSIGNED Public	2	L2_EYE_LEN	object length
(A)	UNSIGNED Public	2	L2_EYE_OFFSET	offset of eye-catcher in object
(C)	CHAR Public	12	L2_EYE_STRING	'>DFHL2xxxxxx'

Offset Hex	Type	Len	Name (Dim)	Description
(18)	CHAR Protected	26	MVS_STREAM_NAME	MVS logstream name
(32)	CHAR Protected	8	JOURNAL_NAME	journal name
(3A)	UNSIGNED Protected	1	LOG_TYPE	log type - MVS or SMF
(3B)	FIXED Protected	1	CONNECTED	connected?
(3C)	FIXED Protected	1	SYSTEM_LOG	CICS system log ind
(3D)	FIXED Protected	1	DASD_ONLY_FLAG	DASD only flag
(3E)	CHAR Protected	16	STRUCTURE_NAME	Structure name
(4E)	CHAR Protected	2	*	reserved
(50)	SIGNED Protected	4	RETENTION_ PERIOD	Retention period
(54)	FIXED Protected	1	AUTO_DELETE_ FLAG	Auto delete flag
(55)	CHAR Protected	3	*	reserved
(58)	FIXED Protected	4	MAX_BLOCK_SIZE	max log block size
(5C)	STRUCTURE IsA(HSMVSSTREAMTOKEN) Protected	16	MVS_STREAM_ TOKEN	MVS Logger token
(6C)	ADDRESS Protected	4	BUFFER_PTR	write buffer ptr
(70)	FIXED Protected	4	BUFFER_LEN	write buffer length
(74)	FIXED Protected	4	WRITE_ECB	ECB for writing block
(78)	STRUCTURE IsA(HSANSAREA) Protected	40	WRITE_ANSA	ixgwrite answer area
(A0)	CHAR Protected	8	CUR_BLOCK_ID	block id
(A8)	CHAR Protected	16	CUR_TIMESTAMP	block timestamp
(A8)	CHAR Protected	8	CUR_TIME_GMT	GMT time
(B0)	CHAR Protected	8	CUR_TIME_LOCAL	local time
(B8)	FIXED Protected	1	MSL_WARNING_MSG	warning msg issued
(B9)	FIXED Protected	1	BROKEN_LOG	log in error flag
(BA)	CHAR Protected	2	*	reserved
(BC)	FIXED Protected	4	BROKEN_RSP	broken response
(C0)	FIXED Protected	4	BROKEN_RSN	broken reason
(C4)	FIXED Protected	4	SMF_RESPONSE	SMF write response
(C8)	FIXED Protected	4	SMF_REASON	SMF write reason
(CC)	CHAR Protected	29	LOG_STREAM_ STATS	various statistics
(CC)	SIGNED Protected	4	IXGWRITE_COUNT	no of writes
(D0)	BITSTRING Protected	8	IXGWRITE_BYTES	no of bytes written
(D8)	SIGNED Protected	4	RETRY_ERRCOUNT	no of retryable errors
(DC)	SIGNED Protected	4	IXGBROST_COUNT	no of browse starts
(E0)	SIGNED Protected	4	IXGBRORD_COUNT	no of browse reads
(E4)	SIGNED Protected	4	IXGDELET_COUNT	no of deletes
(E8)	FIXED Protected	1	RETRY_ERRCOUNT_ INC_DONE	to ensure stats only incremented once
(E9)	CHAR Protected	7	*	reserved
(F0)	CHAR Protected	8	IXG_STCK	Timestamp of last call
(F8)	CHAR Protected	8	IXGWRITE_STCK	IXGWRITE timestamp
(100)	UNSIGNED Protected	4	IXGWRITE_ LATENCY	IXGWRITE latency
(104)	CHAR Protected	20	*	reserved

--
-

Declare asociated types for HardStream.

SHARED DATA
Declared Data

(0)	CHAR Public	4	HSREADTOKEN
(0)	FIXED Public	4	HSLENGTHBYTES
(0)	CHAR Public	16	HSMVSSTREAMTOKEN
(0)	FIXED Public	4	HSECB
(0)	CHAR Protected	40	HSANSAREA
(0)	FIXED Protected	4	HSRETRSN

Constants

Len	Type	Value	Name	Description
4	DECIMAL	1073741824	ECB_POSTED	
4	DECIMAL	0	ECB_CLEAR	
4	DECIMAL	3000	MAX_TRACE_ BLOCK_LEN	
8	CHARACTER	LGWRITE	WAIT_RESOURCE_ TYPE_WRITE	
4	DECIMAL	1	LOST_ACCESS	
4	DECIMAL	2	LOST_DATA	
4	DECIMAL	3	IO_IN_PROGRESS	
4	DECIMAL	4	CONNECT_FAILURE	
4	DECIMAL	5	LOG_NOT_DEFINED	
4	DECIMAL	6	EMPTY_LOG_STREAM	
4	DECIMAL	7	NO_DATA	
4	DECIMAL	72	QBUF_LENGTH	
4	DECIMAL	0	QBUFVERNUM	

L2LF Log manager log formats

-

What follows declares the types that are used when building log blocks and records, and any associated constants.

A block is the unit by which data is written to and read from a logstream. Each block comprises a block header followed by a number of records. Each record comprises a record header followed by caller data. Records on the system log additionally contain links to other records on the logstream. The links are known as chain headers.

There are type declarations for block headers, record headers, chain headers, and the flattened form of a record token, plus the simple types that comprise them.

--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	9	FLATBLOCK	
(0)	CHARACTER	8	ID_OR_NUMBER	block id or number
(0)	CHARACTER	8	FLAT_BLOCK_NUM	block number
(0)	CHARACTER	8	FLAT_BLOCK_ID	block id
(8)	CHARACTER	1	BLOCK_ID_USED	block id used = 'Y', block number used = 'N'

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	FLATRECORDTOKEN	
(0)	STRUCTURE IsA(FLATBLOCK)	9	FLAT_BLOCK	block details
(0)	CHARACTER	8	ID_OR_NUMBER	block id or number
(0)	CHARACTER	8	FLAT_BLOCK_NUM	block number
(0)	CHARACTER	8	FLAT_BLOCK_ID	block id
(8)	CHARACTER	1	BLOCK_ID_USED	block id used = 'Y', block number used = 'N'
(9)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(A)	CHARACTER	2	FLAT_RSVD1	reserved
(C)	UNSIGNED	4	FLAT_INDEX	offset within block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	40	MVSLOGBLOCKHEADER	
(0)	CHARACTER	8	LGBH_GLOBAL_INFO	
(0)	CHARACTER	4	LGBH_BLOCK_TYPE	set to '>DFH' to
(0)	CHARACTER	1	LGBH_BT_ARROW	identify a CICS
(1)	CHARACTER	3	LGBH_BT_DFH	block
(4)	CHARACTER	4	*	
(4)	UNSIGNED	1	LGBH_LOG_TYPE	general or system log
(5)	CHARACTER	1	LGBH_FLAGS	reserved
(6)	UNSIGNED	2	LGBH_BLOCK_VER	block format version number
(8)	CHARACTER	24	LGBH_CICS_INFO	
(8)	CHARACTER	8	LGBH_GENERIC_ APPLID	CICS generic applid

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHARACTER	8	LGBH_START_GMT	record time (GMT)
(18)	CHARACTER	8	LGBH_START_ LOCAL	record time (LOCAL)
(20)	CHARACTER	8	LGBH_BLOCK_INFO	
(20)	CHARACTER	8	LGBH_BLOCK_ NUMBER	block sequence number
(28)	CHARACTER		LGBH_DATA	records follow

-

NOTE: +This block should match the definition in DFHSMFDS+.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	158	SMFLOGBLOCKHEADER	
(0)	CHARACTER	44	SMF_HEADER	
(0)	UNSIGNED	2	SMFH_LEN	record length
(2)	UNSIGNED	2	SMFH_SEG	segment descriptor
(4)	CHARACTER	1	SMFH_FLG	operating system indicator (see constant prefixed smfh_flg below)
(5)	CHARACTER	1	SMFH_RTY	record type (see constant prefixed smfh_rty below)
(6)	CHARACTER	4	SMFH_TME	time record moved (HHMMSST+)
(A)	CHARACTER	4	SMFH_DTE	date record moved (0CYYDDD+)
(E)	CHARACTER	4	SMFH_SID	system identification
(12)	CHARACTER	4	SMFH_SSI	sub-system identification (see constant prefixed smfh_ssi below)
(16)	UNSIGNED	2	SMFH_STY	record subtype (see constant prefixed smfh_sty below)
(18)	UNSIGNED	2	SMFH_TRN	number of triplets in record
(1A)	UNSIGNED	2	SMFH_RSVD1	reserved
(1C)	UNSIGNED	4	SMFH_APS	offset to CICS product section
(20)	UNSIGNED	2	SMFH_LPS	length of CICS product section
(22)	UNSIGNED	2	SMFH_NPS	number of CICS product sections
(24)	UNSIGNED	4	SMFH_ASS	offset to CICS data section
(28)	UNSIGNED	2	SMFH_ASL	length of CICS data section
(2A)	UNSIGNED	2	SMFH_ASN	number of CICS data sections
(2C)	CHARACTER		*	
(2C)	CHARACTER	114	SMF_PRODUCT_ SECTION	
(2C)	CHARACTER	2	SMFPS_VRM	record version format x'0vrm' v = version r = release m = modification (set to &SMF in DFHSYS)
(2E)	CHARACTER	8	SMFPS_PRN	product name (generic APPLID)
(36)	CHARACTER	8	SMFPS_SPN	specific APPLID
(3E)	CHARACTER	2	SMFPS_MFL	record maintenance indicator
(40)	CHARACTER	2	SMFPS_RSVD2	reserved
(42)	CHARACTER	52	SMFPS_RSVD3	reserved
(76)	CHARACTER	8	SMFPS_JNM	journal name
(7E)	CHARACTER	8	SMFPS_JBN	jobname
(86)	CHARACTER	4	SMFPS_RSD	job date
(8A)	CHARACTER	4	SMFPS_RST	job time
(8E)	CHARACTER	8	SMFPS_UIF	user identification
(96)	CHARACTER	8	SMFPS_PDN	operating system product level
(9E)	CHARACTER		*	
(9E)	CHARACTER		SMF_DATA_SECTION	CICS records
(9E)	CHARACTER		SMFDS_DATA	records follow

--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	SYSLOGBLOCKHEADER	
(0)	STRUCTURE	40	SLBH IsA(MVSLOGBLOCKHEADER)	
(0)	CHARACTER	8	LGBH_GLOBAL_ INFO	
(0)	CHARACTER	4	LGBH_BLOCK_ TYPE	set to '>DFH' to
(0)	CHARACTER	1	LGBH_BT_ARROW	identify a CICS
(1)	CHARACTER	3	LGBH_BT_DFH	block
(4)	CHARACTER	4	*	
(4)	UNSIGNED	1	LGBH_LOG_TYPE	general or system log
(5)	CHARACTER	1	LGBH_FLAGS	reserved
(6)	UNSIGNED	2	LGBH_BLOCK_ VER	block format version number
(8)	CHARACTER	24	LGBH_CICS_INFO	
(8)	CHARACTER	8	LGBH_GENERIC_ APPLID	CICS generic applid
(10)	CHARACTER	8	LGBH_START_GMT	record time (GMT)
(18)	CHARACTER	8	LGBH_START_ LOCAL	record time (LOCAL)
(20)	CHARACTER	8	LGBH_BLOCK_INFO	
(20)	CHARACTER	8	LGBH_BLOCK_ NUMBER	block sequence number
(28)	CHARACTER		LGBH_DATA	records follow

Offset Hex	Type	Len	Name (Dim)	Description
(28)	CHARACTER	8	SLBH_PREV_ BLOCK_ID	block id prev block
(30)	UNSIGNED	4	SLBH_LAST_ USED_INDEX	index of last record
(34)	CHARACTER		SLBH_DATA	records follow

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	GENLOGRECORD	
(0)	CHARACTER	12	*	
(0)	UNSIGNED	4	GLRH_RECORD_ LENGTH	inclusive length of this record
(4)	UNSIGNED	4	GLRH_HEADER_ LENGTH	inclusive length of this header
(8)	UNSIGNED	4	GLRH_REC_ DATA_LEN	length of data following this header
(C)	CHARACTER	16	GLRH_TIMESTAMPS	timestamps
(C)	CHARACTER	8	GLRH_GMT	record time (GMT)
(14)	CHARACTER	8	GLRH_LOCAL	record time (LOCAL)
(1C)	CHARACTER	12	GLRH_TASK_INFO	logging task information
(1C)	CHARACTER	4	GLRH_TRAN_ID	transaction id
(20)	CHARACTER	4	GLRH_TASK_ID	task number
(24)	CHARACTER	4	GLRH_TERM_ID	terminal id
(28)	CHARACTER	12	GLRH_RECORD_ID	record identification
(28)	UNSIGNED	2	GLRH_REC_TYPE	start_of_run (sor) or user
(2A)	CHARACTER	2	GLRH_REC_COMPID	logging component id
(2C)	CHARACTER	8	GLRH_REC_ JOURNAL	logging journal name
(34)	CHARACTER	4	GLRH_LGSSI	for DFHLGSSI conversion rtn
(34)	CHARACTER	1	GLRH_LGSSI_ FLAGS	not set for system log
	1...		GLRH_START_ OF_TASK	equivalent to JCSPSOTK
	.1..		GLRH_START_ OF_UOW	equivalent to JCSPLSTK
(35)	CHARACTER	3	GLRH_LGSSI_RSVD	reserved
(38)	CHARACTER		GLRH_REC_DATA	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	20	STARTOFRUNDATA	
(0)	CHARACTER	20	SOR_CICS_INFO	start-of-run information
(0)	CHARACTER	4	SOR_CICS_ RELEASE	CICS version and release
(4)	CHARACTER	8	SOR_SPECIFIC_ APPLID	CICS specific applid
(C)	CHARACTER	8	SOR_CICS_ USERNAME	CICS userid

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	SYSLOGRECORD	
(0)	UNSIGNED	4	SLH_P_REC_LEN	inclusive length of this record
(4)	UNSIGNED	4	SLH_P_HDR_LEN	inclusive length of this header
(8)	CHARACTER	8	SLH_P_STCK	record time (GMT)
(10)	CHARACTER		SLH_P_DATA	start of rest of record

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	MASTERCHAINHEADER	
(0)	STRUCTURE	16	MASTER_PREV	previous on master chain
	IsA(FLATRECORDTOKEN)			
(0)	CHARACTER	9	FLAT_BLOCK	block details
(0)	CHARACTER	8	ID_OR_NUMBER	block id or number
(0)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(0)	CHARACTER	8	FLAT_BLOCK_ID	block id
(8)	CHARACTER	1	BLOCK_ID_USED	block id used = 'Y', block number used = 'N'
(9)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(A)	CHARACTER	2	FLAT_RSVD1	reserved
(C)	UNSIGNED	4	FLAT_INDEX	offset within block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	CHAIN_HEADER	has five variants
(0)	UNSIGNED	4	REC_TYPE	see constants below
(4)	CHARACTER		*	variant-specific data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	20	NORMAL_ CHAIN_HEADER	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	UNSIGNED	4	REC_TYPE_NORMAL	normal type (= 1)
(4)	STRUCTURE IsA(FLATRECORDTOKEN)	16	CHAIN_PREV	previous on UOW chain
(4)	CHARACTER	9	FLAT_BLOCK	block details
(4)	CHARACTER	8	ID_OR_NUMBER	block id or number
(4)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(4)	CHARACTER	8	FLAT_BLOCK_ID	block id
(C)	CHARACTER	1	BLOCK_ID_USED	block id used = 'Y', block number used = 'N'
(D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(E)	CHARACTER	2	FLAT_RSVD1	reserved
(10)	UNSIGNED	4	FLAT_INDEX	offset within block
(14)	CHARACTER		NORMAL_RM_START	start of RM data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	36	FORK_CHAIN_HEADER	
(0)	UNSIGNED	4	REC_TYPE_FORK	fork type (= 2)
(4)	STRUCTURE IsA(FLATRECORDTOKEN)	16	CHAIN_PREV_LIVE	previous on UOW chain on secondary
(4)	CHARACTER	9	FLAT_BLOCK	block details
(4)	CHARACTER	8	ID_OR_NUMBER	block id or number
(4)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(4)	CHARACTER	8	FLAT_BLOCK_ID	block id
(C)	CHARACTER	1	BLOCK_ID_USED	block id used = 'Y', block number used = 'N'
(D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(E)	CHARACTER	2	FLAT_RSVD1	reserved
(10)	UNSIGNED	4	FLAT_INDEX	offset within block
(14)	STRUCTURE IsA(FLATRECORDTOKEN)	16	CHAIN_PREV_DEAD	previous on UOW chain on primary
(14)	CHARACTER	9	FLAT_BLOCK	block details
(14)	CHARACTER	8	ID_OR_NUMBER	block id or number
(14)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(14)	CHARACTER	8	FLAT_BLOCK_ID	block id
(1C)	CHARACTER	1	BLOCK_ID_USED	block id used = 'Y', block number used = 'N'
(1D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(1E)	CHARACTER	2	FLAT_RSVD1	reserved
(20)	UNSIGNED	4	FLAT_INDEX	offset within block
(24)	CHARACTER		FORK_RM_START	start of RM data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	20	SECONDARY_ CHAIN_HEADER	
(0)	UNSIGNED	4	REC_TYPE_SEC	secondary type (= 3)
(4)	STRUCTURE IsA(FLATRECORDTOKEN)	16	CHAIN_PREV_SEC	previous on UOW chain
(4)	CHARACTER	9	FLAT_BLOCK	block details
(4)	CHARACTER	8	ID_OR_NUMBER	block id or number
(4)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(4)	CHARACTER	8	FLAT_BLOCK_ID	block id
(C)	CHARACTER	1	BLOCK_ID_USED	block id used = 'Y', block number used = 'N'
(D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(E)	CHARACTER	2	FLAT_RSVD1	reserved
(10)	UNSIGNED	4	FLAT_INDEX	offset within block
(14)	CHARACTER		SECONDARY_ RM_START	start of RM data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	USER_CHAIN_HEADER	
(0)	UNSIGNED	4	REC_TYPE_USER	user type (= 4)
(4)	CHARACTER		USER_RM_START	start of RM data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	36	TRIM_CHAIN_HEADER	
(0)	UNSIGNED	4	REC_TYPE_TRIM	trim type (= 5)
(4)	CHARACTER	16	PRIMARY_ LOG_HISTORY_ POINT_INFO	to trim primary
(4)	CHARACTER	8	PRIMARY_ STCK_VALUE	store clock value
(C)	CHARACTER	8	PRIMARY_ BLOCK_ID	MVS block id
(14)	CHARACTER	16	SECONDARY_ LOG_HISTORY_ POINT_INFO	to trim secondary
(14)	CHARACTER	8	SECONDARY_ STCK_VALUE	

Offset Hex	Type	Len	Name (Dim)	Description
				store clock value
(1C)	CHARACTER	8	SECONDARY_ BLOCK_ID	MVS block id
(24)	CHARACTER		*	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	20	NON_MOVED_ CHAIN_HEADER	
(0)	UNSIGNED	4	REC_TYPE_NORMAL	normal type (= 6)
(4)	STRUCTURE IsA(FLATRECORDTOKEN)	16	CHAIN_PREV	prev on UOW chain
(4)	CHARACTER	9	FLAT_BLOCK	block details
(4)	CHARACTER	8	ID_OR_NUMBER	block id or number
(4)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(4)	CHARACTER	8	FLAT_BLOCK_ID	block id
(C)	CHARACTER	1	BLOCK_ID_USED	block id used = 'Y', block number used = 'N'
(D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(E)	CHARACTER	2	FLAT_RSVD1	reserved
(10)	UNSIGNED	4	FLAT_INDEX	offset within block
(14)	CHARACTER		NON_MOVED_ RM_START	start of RM data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	68	SYSLOGCOMBINEDRECORD	
(0)	STRUCTURE IsA(SYSLOGRECORD)	16	SLH_PREFIX	initial header
(0)	UNSIGNED	4	SLH_P_REC_LEN	inclusive length of this record
(4)	UNSIGNED	4	SLH_P_HDR_LEN	inclusive length of this header
(8)	CHARACTER	8	SLH_P_STCK	record time (GMT)
(10)	CHARACTER		SLH_P_DATA	start of rest of record
(10)	STRUCTURE IsA(MASTERCHAINHEADER)	16	SLH_MASTER	link to previous
(10)	CHARACTER	16	MASTER_PREV	previous on master chain
(10)	CHARACTER	9	FLAT_BLOCK	block details
(10)	CHARACTER	8	ID_OR_NUMBER	block id or number
(10)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(10)	CHARACTER	8	FLAT_BLOCK_ ID	block id
(18)	CHARACTER	1	BLOCK_ID_USED	block id used = 'Y', block number used = 'N'
(19)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(1A)	CHARACTER	2	FLAT_RSVD1	reserved
(1C)	UNSIGNED	4	FLAT_INDEX	offset within block
(20)	CHARACTER	36	SLH_REST	record is one of...
(20)	STRUCTURE IsA(NORMAL_CHAIN_HEADER)	20	SLH_NORMAL	normal primary
(20)	UNSIGNED	4	REC_TYPE_ NORMAL	normal type (= 1)
(24)	CHARACTER	16	CHAIN_PREV	previous on UOW chain
(24)	CHARACTER	9	FLAT_BLOCK	block details
(24)	CHARACTER	8	ID_OR_NUMBER	block id or number
(24)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(24)	CHARACTER	8	FLAT_BLOCK_ ID	block id
(2C)	CHARACTER	1	BLOCK_ID_ USED	block id used = 'Y', block number used = 'N'
(2D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(2E)	CHARACTER	2	FLAT_RSVD1	reserved
(30)	UNSIGNED	4	FLAT_INDEX	offset within block
(34)	CHARACTER		NORMAL_ RM_START	start of RM data
(20)	STRUCTURE IsA(FORK_CHAIN_HEADER)	36	SLH_FORK	fork
(20)	UNSIGNED	4	REC_TYPE_FORK	fork type (= 2)
(24)	CHARACTER	16	CHAIN_PREV_ LIVE	previous on UOW chain on secondary
(24)	CHARACTER	9	FLAT_BLOCK	block details
(24)	CHARACTER	8	ID_OR_NUMBER	block id or number
(24)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(24)	CHARACTER	8	FLAT_BLOCK_ ID	block id
(2C)	CHARACTER	1	BLOCK_ID_ USED	block id used = 'Y', block number used = 'N'
(2D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(2E)	CHARACTER	2	FLAT_RSVD1	reserved
(30)	UNSIGNED	4	FLAT_INDEX	offset within block
(34)	CHARACTER	16	CHAIN_PREV_ DEAD	previous on UOW chain on primary
(34)	CHARACTER	9	FLAT_BLOCK	block details
(34)	CHARACTER	8	ID_OR_NUMBER	block id or number
(34)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(34)	CHARACTER	8	FLAT_BLOCK_ ID	block id
(3C)	CHARACTER	1	BLOCK_ID_ USED	block id used = 'Y', block number used = 'N'
(3D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(3E)	CHARACTER	2	FLAT_RSVD1	reserved
(40)	UNSIGNED	4	FLAT_INDEX	offset within block
(44)	CHARACTER		FORK_RM_START	start of RM data

Offset Hex	Type	Len	Name (Dim)	Description
(20)	STRUCTURE	20	SLH_SECONDARY	secondary
	IsA(SECONDARY_CHAIN_HEADER)			
(20)	UNSIGNED	4	REC_TYPE_SEC	secondary type (= 3)
(24)	CHARACTER	16	CHAIN_PREV_SEC	previous on UOW chain
(24)	CHARACTER	9	FLAT_BLOCK	block details
(24)	CHARACTER	8	ID_OR_NUMBER	block id or number
(24)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(24)	CHARACTER	8	FLAT_BLOCK_ ID	block id
(2C)	CHARACTER	1	BLOCK_ID_ USED	block id used = 'Y', block number used = 'N'
(2D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(2E)	CHARACTER	2	FLAT_RSVD1	reserved
(30)	UNSIGNED	4	FLAT_INDEX	offset within block
(34)	CHARACTER		SECONDARY_RM_START	start of RM data
(20)	STRUCTURE	4	SLH_USER	unchained user
	IsA(USER_CHAIN_HEADER)			
(20)	UNSIGNED	4	REC_TYPE_USER	user type (= 4)
(24)	CHARACTER		USER_RM_START	start of RM data
(20)	STRUCTURE	36	SLH_TRIM	unchained trim
	IsA(TRIM_CHAIN_HEADER)			
(20)	UNSIGNED	4	REC_TYPE_TRIM	trim type (= 5)
(24)	CHARACTER	16	PRIMARY_LOG_HISTORY_POINT_INFO	to trim primary
(24)	CHARACTER	8	PRIMARY_STCK_VALUE	store clock value
(2C)	CHARACTER	8	PRIMARY_ BLOCK_ID	MVS block id
(34)	CHARACTER	16	SECONDARY_LOG_HISTORY_POINT_INFO	to trim secondary
(34)	CHARACTER	8	SECONDARY_STCK_VALUE	store clock value
(3C)	CHARACTER	8	SECONDARY_BLOCK_ID	MVS block id
(44)	CHARACTER		*	
(20)	STRUCTURE	20	SLH_NON_MOVED	1ry
	IsA(NON_MOVED_CHAIN_HEADER)			
(20)	UNSIGNED	4	REC_TYPE_ NORMAL	normal type (= 6)
(24)	CHARACTER	16	CHAIN_PREV	prev on UOW chain
(24)	CHARACTER	9	FLAT_BLOCK	block details
(24)	CHARACTER	8	ID_OR_NUMBER	block id or number
(24)	CHARACTER	8	FLAT_BLOCK_ NUM	block number
(24)	CHARACTER	8	FLAT_BLOCK_ ID	block id
(2C)	CHARACTER	1	BLOCK_ID_ USED	block id used = 'Y', block number used = 'N'
(2D)	CHARACTER	1	FLAT_REAL	real record token = 'Y', null record token = 'N'
(2E)	CHARACTER	2	FLAT_RSVD1	reserved
(30)	UNSIGNED	4	FLAT_INDEX	offset within block
(34)	CHARACTER		NON_MOVED_RM_START	start of RM data
(44)	CHARACTER		*	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	GENLOGUSER	
(0)	UNSIGNED	4	CL_UH_LENGTH	length of structure inclusive of this field
(4)	UNSIGNED	2	CL_UH_JOURNAL_ TYPE	journal type
(6)	CHARACTER	2	CL_UH_RSVD1	reserved
(8)	UNSIGNED	4	CL_UH_PREFIX_ LENGTH	user prefix length
(C)	CHARACTER		CL_UH_END	user prefix data (if any) followed by user data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	SYSLOGUSER	
(0)	CHARACTER	16	SL_UH_TRAN_DATA	
(0)	UNSIGNED	4	SL_UH_TD_LENGTH	length of this header
(4)	CHARACTER	4	SL_UH_TD_TASKNO	task number
(8)	CHARACTER	4	SL_UH_TD_TRANID	tranid
(C)	CHARACTER	4	SL_UH_TD_TERMID	termid
(10)	CHARACTER		SL_UH_END	general user header follows

Constants

Len	Type	Value	Name	Description
2	DECIMAL	1	LGBH_BLOCK_ VERSION_NO	
3	CHARACTER	DFH	LGBH_BLOCK_ TYPE_DFH	
1	CHARACTER	>	LGBH_BLOCK_ TYPE_ARROW	
1	DECIMAL	0	LGBH_LOG_ TYPE_GENERAL	
1	DECIMAL	1	LGBH_LOG_ TYPE_SYSTEM	
2	DECIMAL	1	SOR_REC_TYPE	
2	DECIMAL	2	USER_REC_TYPE	
2	DECIMAL	1	SLBH_BLOCK_ VERSION_NO	
3	CHARACTER	DFH	SLBH_BLOCK_ TYPE_DFH	
1	CHARACTER	>	SLBH_BLOCK_ TYPE_ARROW	
1	DECIMAL	0	SLBH_LOG_ TYPE_GENERAL	
1	DECIMAL	1	SLBH_LOG_ TYPE_SYSTEM	
4	DECIMAL	1	SLH_P_REC_ TYPE_NORMAL	
4	DECIMAL	2	SLH_P_REC_ TYPE_FORK	
4	DECIMAL	3	SLH_P_REC_ TYPE_SECONDARY	
4	DECIMAL	4	SLH_P_REC_ TYPE_USER	
4	DECIMAL	5	SLH_P_REC_ TYPE_TRIM	
4	DECIMAL	6	SLH_P_REC_ TYPE_NON_MOVED	
4	CHARACTER	CICS	SMFH_SSI_CICS	sub-system identification
1	CHARACTER	ú	SMFH_FLG_ESA4	MVS/ESA V4
1	CHARACTER	>	SMFH_RTY_110	record type 110 for CICS
2	DECIMAL	0	SMFH_STY_LG	for journaling
2	DECIMAL	1	SMFH_STY_MN	for monitoring
2	DECIMAL	2	SMFH_STY_ST	for statistics
4	DECIMAL	2	SMFH_NUMBER_ TRIPLETS	
4	DECIMAL	0	SMFH_MFL_ID	
2	HEX	0530	SMFPS_VRM_VAL	
2	DECIMAL	0	SMFPS_MFL_0	
4	DECIMAL	44	SMFH_PRD_ SECT_OFFSET	
4	DECIMAL	114	SMFH_PRD_ SECT_LENGTH	
4	DECIMAL	1	SMFH_PRD_ SECT_NUMBER	
4	DECIMAL	158	SMFH_DATA_ SECT_OFFSET	
4	DECIMAL	0	SMFH_DATA_ SECT_LENGTH	
4	DECIMAL	1	SMFH_DATA_ SECT_NUMBER	
4	DECIMAL	32756	SMF_MAX_BLOCK_LEN	
4	DECIMAL	32598	SMF_MAX_DATA_ SECTION_LEN	

L2LT Log manager lock tracker class

-

What follows defines the Log Manager LockTracker class.

Several Log Manager objects contain a lock. Such objects are Chains, Streams and Domain Manager. Under certain circumstances, notably when its recovery routine has been driven, a module that uses such an object needs to know whether a method it called has acquired the lock. This is so the lock can be released. It is therefore necessary to track the status of the lock. This requires knowing both the address of the object and whether the lock is held or not.

This is achieved by declaring a LockTracker variable for each object lock the module is interested in. Each LockTracker must be explicitly initialised by the module using the l2lt_set_free method. Whenever the lock is acquired or released the LockTracker is automatically updated by the object using the l2lt_set_held and l2lt_set_free methods. If the module recovery routine is driven it must call the lock_release method of the object. This uses the l2lt_inq_status and l2lt_inq_token methods, and will only release the lock if the LockTracker indicates the lock is held.

Only one Chain lock, one Stream lock and the Domain Manager lock may be tracked within a given module. This is because a LockTracker is not passed as a parameter to Stream or Chain.

-

The LockTracker class has instance data but no class data.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	8	LOCKTRACKER	

-

An instance of the LockTracker class consists of a token to identify the object in question, plus the status of the lock.

INSTANCE DATA

Declared Data

(0)	CHAR Protected	8	INSTANCE_ DATA_BLOCK	
(0)	ADDRESS Protected	4	OBJECT_TOKEN	locates the object
(4)	BITSTRING Protected	1	LOCK_STATUS	object lock status
	`1... .... Protected`		HELD	
	`.111 1111 Protected`		*	
(5)	CHAR Protected	3	*	reserved

L2RT Log manager record token class

-

What follows defines the Log Manager RecordToken class.

A RecordToken provides a means of identifying the location of a log record that is being written to or read from a logstream. It consists of a pointer to the Block object for the block containing the record, and an index which gives the offset of the record within that block.

A 'flattened' form of a RecordToken is also required, so that the information contained within a RecordToken may be stored in log records, and later unflattened when the record is read back. The FlatRecordToken is defined with the log formats in DFHL2LFC.

Whenever a RecordToken is created (by building, copying or unflattening) we immediately register interest in it. This holds the Block, and means that the Block can not disappear from under our caller's feet. When our caller has finished with the RecordToken he must deregister interest, and we will release the hold on the Block. Releasing the last hold destroys the Block.

-

The RecordToken class has instance data but no class data.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	24	RECORDTOKEN	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	

-

An instance of the RecordToken class consists of a pointer to the associated Block object, and an index which is the offset of the record within that block. Note that the largest size block that MVS allows is 64K bytes.

A null RecordToken has no underlying Block and so has a null pointer and an index of zero.

Offset Hex	Type	Len	Name (Dim)	Description
Declared Data				
(8)	CHAR Protected	10	INSTANCE_ DATA_BLOCK	
(8)	ADDRESS Protected	4	BLOCK_PTR	pointer to Block object
(C)	UNSIGNED Protected	4	INDEX	offset within block
(10)	CHAR Protected	2	*	reserved

L2SL Log manager system log class

-

What follows defines the Log Manager SystemLog class.

The CICS system log consists of two MVS Logger logstreams, the primary (journal name DFHLOG) and the secondary (journal name DFHSHUNT). The SystemLog class knows which log stream objects are used for these (that is, which instances of the BrowseableStream class). It is responsible for opening the log streams at CICS startup, and for deleting all records from the log streams when CICS is cold started. It provides inquiry methods so other classes can obtain the tokens (actually BrLogStreamTokens) for the primary and secondary streams.

It is possible for the user to define the primary and/or secondary stream as a dummy stream. If the primary is a dummy then this implies that the secondary is also a dummy (it does not make sense otherwise). A special dummy BrLogStreamToken is used to indicate that a stream is a dummy, and is returned by the inquiry method. It is the inquirer that decides upon the appropriate action to take.

The SystemLog class owns the activity keypoint frequency (AKPFREQ). It provides methods for inquiring and setting its value. It also passes on the value of the activity keypoint frequency to the primary stream object. The activity keypoint frequency can be set at CICS startup and using the CICS API. If it is set at CICS startup and if the primary stream has not yet been opened, the call to the primary stream object is deferred until the open takes place.

The SystemLog class must be notified of any failures that occur when writing critical data to or reading critical data from the primary or secondary stream. This normally results in a termination of CICS.

-

The SystemLog class has no instance data as there are no instances of this class. All data is stored in class data and is accessed by class methods. It has both internal and external methods.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	4	SYSTEMLOG	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	

-

The SystemLog class data consists of the tokens for the primary and secondary streams, the activity keypoint frequency, the inhibit delete indicator, some deferred event indicators used when opening and deleting all records from the secondary stream and when passing on the activity keypoint frequency, and a flag that is set to indicate CICS is quiescing due to a lost data failure.

The BrLogStreamToken for each stream can take one of the following values:

- Null - the stream has not been opened

- Dummy - the stream is defined as a dummy

- A real BrLogStreamToken - the stream is real and was successfully opened

Offset Hex	Type	Len	Name (Dim)	Description
SHARED DATA				
Declared Data				
(0)	CHAR Protected	42	CLASSDATABLOCK	
(0)	STRUCTURE IsA(L2_EYE_CATCHER) Protected	16	EYE_CATCHER	an eye-catcher
(0)	UNSIGNED Public	2	L2_EYE_LEN	object length
(2)	UNSIGNED Public	2	L2_EYE_OFFSET	offset of eye-catcher in object

Offset Hex	Type	Len	Name (Dim)	Description
(4)	CHAR Public	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(10)	ADDRESS Protected	4	PRIMARY_STOKEN	token for primary
(14)	ADDRESS Protected	4	SECONDARY_ STOKEN	token for secondary
(18)	UNSIGNED Protected	4	AKP_FREQUENCY	keypoint frequency
(1C)	BITSTRING Protected	1	DEFER	deferred event flags
	`1... .... Protected`		OPEN_SECONDARY	open secondary
	`.1.. .... Protected`		DELETE_ SECONDARY	delete all secondary
	`..1. .... Protected`		PASS_AKP	pass akp frequency
	`...1 1111 Protected`		*	reserved
(1D)	FIXED Protected	1	QUIESCING	CICS is quiescing?
(1E)	CHAR Protected	12	*	reserved APARs

--
-

Declare associated types. There is a type for the different failures that can occur to the system log, and a type for the different system log operations.

(0)	FIXED Public	1	SYSLOGFAILURE
(0)	FIXED Public	1	SYSLOGOPERATION

Constants

Len	Type	Value	Name	Description
8	CHARACTER	DFHLOG	SL_PRIMARY	
8	CHARACTER	DFHSHUNT	SL_SECONDARY	
4	DECIMAL	0	NULL_LOGSTREAM_ TOKEN	
4	DECIMAL	1	DUMMY_LOGSTREAM_ TOKEN	
1	DECIMAL	0	SLF_NONE	
1	DECIMAL	1	SLF_LOST_DATA	
1	DECIMAL	2	SLF_LOST_ACCESS	
1	DECIMAL	3	SLF_BAD_BLOCK_SIZE	
1	DECIMAL	4	SLF_DISASTER	
1	DECIMAL	5	SLF_DATA_NOT_FOUND	
1	DECIMAL	6	SLF_NOT_ACTIVE	
1	DECIMAL	1	SLO_WRITE	
1	DECIMAL	2	SLO_READ	
1	DECIMAL	3	SLO_RESTART	
4	DECIMAL	200	AKP_MIN	
4	DECIMAL	65535	AKP_MAX	
4	DECIMAL	1	OUT_OF_RANGE	

L2SR Log manager stream class

-

What follows defines the Log Manager Stream class.

A Stream object provides the ability to write data records to and read data records from an MVS Logger or SMF logstream. It provides a layer between the logstream user and the code that actually calls MVS. This layer is necessary to hide the details involved with writing to and reading from logstreams. In particular, it provides a record-level interface for the logstream user, and it hides various performance related techniques such as double buffering and deferred force of buffers.

A logstream may be viewed as consisting of a number of blocks. These are the units by which data is written to the physical medium. A logstream will typically comprise a number of such blocks on the physical medium (referred to as +hard+), plus two buffers called +Current+ and +Previous+ which provide the double buffering when writing data (referred to as +soft+), plus possibly some +Read+ buffers used when reading blocks back from the logstream.

A Block object represents an individual block on the hard stream or a buffer. A Stream object therefore cooperates with several Block objects when writing and reading data. However a Block is not independent of the Stream that it belongs to. A Block object requires some context information, primarily to implement its block numbering scheme. This context data is owned by Block, is held as part of a Stream object, and is passed to Block methods where appropriate.

A General Log logstream is represented by a Stream object. However, a System Log logstream is more complex and is represented by a BrowseableStream object. The BrowseableStream class inherits from the Stream class, and so has all the properties of Stream declared here.

-

The Stream class has both instance and class data. It has both internal and external methods.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	608	STREAM	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	

Offset Hex	Type	Len	Name (Dim)	Description
-				

An instance of Stream class consists of:

- An eyecatcher.

- A double chain link to other streams in the chain of all streams.

- A stream lock which is used to manage concurrent requests made against the stream. Note that a Stream method requiring both the stream lock and the domain lock should acquire the stream lock first to prevent possible deadlock.

- Two block-oriented data structures called StreamBlocks used for managing writes and deferred writes. At any given time one is for the Current block and the other is for the Previous block.

- Pointers to the two StreamBlocks above. One identifies the Current, the other identifies the Previous.

- The ForceToken currently associated with this stream. This is updated on every buffer switch.

- The activity keypoint frequency of the stream, set to zero if activity keypoints do not apply, and an associated count which is used to monitor when activity keypoints are to be triggered.

- Some context data which is owned by the Block class, and is passed to those Block methods that require it.

- The HardStream object that is associated with this stream.

- Whether the stream is an MVS Logger log or an SMF log.

- The logstream name. This is for MVS Logger logs only.

- The journal name. This is a real journal name for SMF logs, or is fabricated from the last qualifier of the logstream name for MVS Logger logs.

- Whether the stream is for a System Log or General Log.

- Some flags indicating progress through the initialization of a Stream object.

- A flag indicating whether the deferred flush mechanism is active for the stream.

- Various statistics for monitoring the number of tasks forced to wait while writing to the stream.

Declared Data

(8)	CHAR Protected	600	STREAM_ INSTANCE_DATA	
(8)	STRUCTURE IsA(L2_EYE_CATCHER) Protected	16	EYE_CATCHER	an eye-catcher
(8)	UNSIGNED Public	2	L2_EYE_LEN	object length
(A)	UNSIGNED Public	2	L2_EYE_OFFSET	offset of eye-catcher in object
(C)	CHAR Public	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(18)	OBJECT IsA(HOP_DCHAINNODE) Protected	16	STREAM_ CHAIN_LINK	link in global chain
(18)	CHAR Private	4	*	
(20)	CHAR Protected	8	*	
(20)	ADDRESS Protected	4	PREV	
(24)	ADDRESS Protected	4	NEXT	
(28)	UNSIGNED Protected	4	STREAM_ FORCE_TOKEN	
				Current force token
(2C)	ADDRESS Protected	4	LOCK_TOKEN	stream lock token
(30)	ADDRESS Protected	4	CURRENT	-> Current details
(34)	ADDRESS Protected	4	PREVIOUS	-> Previous details
(38)	STRUCTURE IsA(STREAMBLOCK) Protected	64	FIRST_BLOCK	Curr or Prev details
(38)	ADDRESS Protected	4	BLOCK_PTR	-> actual Block object

Offset Hex	Type	Len	Name (Dim)	Description
(3C)	UNSIGNED Protected	4	FORCE_TOKEN	force token for block
(40)	ADDRESS Protected	4	NEXT_BLOCK_PTR	-> next Block to be Current
(44)	CHAR Protected	4	BLOCK_OWNER	tran number of nominal owner
(48)	CHAR Protected	40	SUSPEND_QUEUE	chain of suspended tasks
(48)	CHAR Private	4	*	
(50)	CHAR Protected	16	ITER0	
(50)	CHAR Private	4	*	
(58)	CHAR Protected	8	*	
(58)	ADDRESS Protected	4	PREV	
(5C)	ADDRESS Protected	4	NEXT	
(60)	CHAR Protected	16	NODE0	
(60)	CHAR Private	4	*	
(68)	CHAR Protected	8	*	
(68)	ADDRESS Protected	4	PREV	
(6C)	ADDRESS Protected	4	NEXT	
(70)	UNSIGNED Protected	1	STATUS	current status
(71)	CHAR Protected	7	*	
(78)	STRUCTURE IsA(STREAMBLOCK) Protected	64	SECOND_BLOCK	Curr or Prev details
(78)	ADDRESS Protected	4	BLOCK_PTR	-> actual Block object
(7C)	UNSIGNED Protected	4	FORCE_TOKEN	force token for block
(80)	ADDRESS Protected	4	NEXT_BLOCK_PTR	-> next Block to be Current
(84)	CHAR Protected	4	BLOCK_OWNER	tran number of nominal owner
(88)	CHAR Protected	40	SUSPEND_QUEUE	chain of suspended tasks
(88)	CHAR Private	4	*	
(90)	CHAR Protected	16	ITER0	
(90)	CHAR Private	4	*	
(98)	CHAR Protected	8	*	
(98)	ADDRESS Protected	4	PREV	
(9C)	ADDRESS Protected	4	NEXT	
(A0)	CHAR Protected	16	NODE0	
(A0)	CHAR Private	4	*	
(A8)	CHAR Protected	8	*	
(A8)	ADDRESS Protected	4	PREV	
(AC)	ADDRESS Protected	4	NEXT	
(B0)	UNSIGNED Protected	1	STATUS	current status
(B1)	CHAR Protected	7	*	
(B8)	UNSIGNED Protected	4	AKP_FREQUENCY	activity keypoint frequency
(BC)	SIGNED Protected	4	AKP_COUNT	take keypoint when count reaches zero
(C0)	CHAR Protected	5	BACKTRACK	progress flags
(C0)	FIXED Protected	1	LOCK_ADDED	stream lock added?
(C1)	FIXED Protected	1	CHAINED	on global chain?
(C2)	FIXED Protected	1	CONNECTED	got hard stream?
(C3)	FIXED Protected	1	GOT_BLOCKS	got Curr and Prev?
(C4)	FIXED Protected	1	STATS_OK	gather stats?
(C5)	FIXED Protected	1	LOST_DATA_ WARNING	lost data signalled?
(C6)	FIXED Protected	1	SYSLOG	system log?
(C7)	UNSIGNED Protected	1	TYPE_OF_STREAM	MVS Logger or SMF?
(C8)	CHAR Protected	8	STREAM_JOURNAL	journal name
(D0)	STRUCTURE IsA(BLOCKCONTEXT) Protected	32	BLOCK_CONTEXT	block context data owned by Block class
(D0)	CHAR Public	8	CURR_BLOCK_NUM	block number of last block created
(D8)	CHAR Public	8	LAST_BLOCK_ID	block id of last block written to MVS
(E0)	CHAR Public	8	LAST_BLOCK_ TIME	creation time of last block written to MVS
(E8)	UNSIGNED Public	1	*	
(E9)	UNSIGNED Public	1	*	
(EA)	CHAR Public	6	*	

Offset Hex	Type	Len	Name (Dim)	Description
(F0)	CHAR Public		*	
(F0)	OBJECT IsA(HARDSTREAM) Protected	280	HARD_STREAM	HardStream object
(F0)	CHAR Private	4	*	

Offset Hex	Type	Len	Name (Dim)	Description

-

An instance of HardStream class consists of

- An eyecatcher.

This helps dump navigation.

- A log stream name.

This is the log stream name which denotes the MVS System Logger log stream on connect operation, which returns a log stream token.

- A journal name.

This is the journal name from the log stream name, used as the resource name when a task is suspended.

- A log type.

This is either 'mvs' or 'smf'.

- A connected/disconnected indicator.

When 'connected' the HardStream object is operational, and when 'disconnected' it has been disconnected and it about to be destroyed.

- A System Log indicator.

If 'Y' the log stream forms part of the System Log.

- dasd_ only(y/n)

This flag indicates whether the log stream is of type DASDONLY or CF based.

- structname

If the log stream is CF based, this is the structure name used by the log stream, otherwise this is set to binary 0 (meaning not applicable).

- retention_ period

The log stream retention period is the number in days that the data must be kept before it can be physically deleted by the MVS logger.

- auto_delete

Auto delete flag, if set to yes the MVS logger automatically deletes the data as it matures beyond the retention period, irrespective of any IXGDELET calls. If set to no the data is deleted when it matures beyond the retention period and an IXGDELET call has been issued.

- A maximum block size.

This is a constant, being the maximum block size allowed for the MVS System Logger log stream or MVS SMF log.

- An MVS log stream token.

This is the token that denotes the MVS Logger log stream at its interface. The MVS System Logger returns this value on the connect operation.

- A buffer pointer.

This is the address of the buffer to be written. It is kept here because of the possibility of the need to retry later due to a recoverable error returned from MVS Logger.

- A buffer length.

This is the length of the buffer to be written. It is kept here because of the possibility of the need to retry later due to a recoverable error returned from MVS Logger.

- An ECB.

This is the ECB used when writing to the MVS Logger log stream or MVS SMF log.

- A write answer area.

This is the area where the MVS Logger returns its asynchronous response and diagnostic data.

- A block id.

This is the area where the MVS Logger returns the block id of the block just written.

- A block timestamp.

This is the area where the MVS System Logger returns timestamp of the block just written.

- Warning received indicator.

Set to 'Y' on receipt of a warning exception from the MVS Logger. Reset to 'N' on the first 'ok' response following the warning. Used to limit the number of times a warning message is issued.

- Broken log indicator.

Set to 'Y' on receipt of an unrecoverable error from the MVS Logger. Maintains this state until the log is disconnected. Subsequent calls to a broken log will receive the same response as the original failure, which are kept in the broken response and reason fields.

- Broken response.

- Broken reason.

- SMF response.

This field is the internal response of an SMF write.

- SMF reason.

This field is the internal reason of an SMF write.

- Various statistics.

These are the stats fields that HardStream supports, which are incremented when appropriate and reported/reset on request. For SMF type log streams all stats fields are not used.

- ixg_stck

This is set to the current STCK value just before calling the MVS logger. This is used by the heartbeat task to determine whether it is appropriate to 'touch' the MVS logger.

- ixgwrite_ stck

This is set to the current STCK value just before calling the MVS logger macro IXGWRITE. This is used to evaluate the IXGWRITE latency.

- ixgwrite_ latency

This is set to the time it took to execute the last IXGWRITE call. If the call is made SYNCronously then this is simply the time taken to execute the call and return. If the call is made ASYNCronously then this includes the initial plus the wait period to the posting of the ECB. This is used to cap the LG defer period. This is measured in miliseconds.

Offset Hex	Type	Len	Name (Dim)	Description
(F8)	CHAR Protected	272	INSTANCE_DATA_BLOCK	
(F8)	CHAR Protected	16	EYE_CATCHER	an eye-catcher
(F8)	UNSIGNED Public	2	L2_EYE_LEN	object length
(FA)	UNSIGNED Public	2	L2_EYE_ OFFSET	offset of eye-catcher in object
(FC)	CHAR Public	12	L2_EYE_ STRING	'>DFHL2xxxxxx'
(108)	CHAR Protected	26	MVS_STREAM_ NAME	MVS logstream name
(122)	CHAR Protected	8	JOURNAL_NAME	journal name

Offset Hex	Type	Len	Name (Dim)	Description
(12A)	UNSIGNED Protected	1	LOG_TYPE	log type - MVS or SMF
(12B)	UNSIGNED Protected	1	CONNECTED	connected?
(12C)	UNSIGNED Protected	1	SYSTEM_LOG	CICS system log ind
(12D)	UNSIGNED Protected	1	DASD_ONLY_ FLAG	DASD only flag
(12E)	CHAR Protected	16	STRUCTURE_ NAME	Structure name
(13E)	CHAR Protected	2	*	
(140)	SIGNED Protected	4	RETENTION_ PERIOD	Retention period
(144)	UNSIGNED Protected	1	AUTO_DELETE_ FLAG	Auto delete flag
(145)	CHAR Protected	3	*	
(148)	UNSIGNED Protected	4	MAX_BLOCK_ SIZE	max log block size
(14C)	CHAR Protected	16	MVS_STREAM_ TOKEN	MVS Logger token
(15C)	ADDRESS Protected	4	BUFFER_PTR	write buffer ptr
(160)	UNSIGNED Protected	4	BUFFER_LEN	write buffer length
(164)	UNSIGNED Protected	4	WRITE_ECB	ECB for writing block
(168)	CHAR Protected	40	WRITE_ANSA	ixgwrite answer area
(190)	CHAR Protected	8	CUR_BLOCK_ID	block id
(198)	CHAR Protected	16	CUR_TIMESTAMP	block timestamp
(198)	CHAR Protected	8	CUR_TIME_GMT	GMT time
(1A0)	CHAR Protected	8	CUR_TIME_ LOCAL	local time
(1A8)	UNSIGNED Protected	1	MSL_WARNING_ MSG	warning msg issued
(1A9)	UNSIGNED Protected	1	BROKEN_LOG	log in error flag
(1AA)	CHAR Protected	2	*	
(1AC)	SIGNED Protected	4	BROKEN_RSP	broken response
(1B0)	SIGNED Protected	4	BROKEN_RSN	broken reason
(1B4)	SIGNED Protected	4	SMF_RESPONSE	SMF write response
(1B8)	SIGNED Protected	4	SMF_REASON	SMF write reason
(1BC)	CHAR Protected	29	LOG_STREAM_ STATS	various statistics
(1BC)	SIGNED Protected	4	IXGWRITE_ COUNT	no of writes
(1C0)	BITSTRING Protected	8	IXGWRITE_ BYTES	no of bytes written
(1C8)	SIGNED Protected	4	RETRY_ERRCOUNT	no of retryable errors
(1CC)	SIGNED Protected	4	IXGBROST_ COUNT	no of browse starts
(1D0)	SIGNED Protected	4	IXGBRORD_ COUNT	no of browse reads
(1D4)	SIGNED Protected	4	IXGDELET_ COUNT	no of deletes
(1D8)	UNSIGNED Protected	1	RETRY_ERRCOUNT_ INC_DONE	to ensure stats only incremented once
(1D9)	CHAR Protected	7	*	
(1E0)	CHAR Protected	8	IXG_STCK	Timestamp of last call
(1E8)	CHAR Protected	8	IXGWRITE_STCK	IXGWRITE timestamp
(1F0)	UNSIGNED Protected	4	IXGWRITE_ LATENCY	IXGWRITE latency
(1F4)	CHAR Protected	20	*	
(208)	CHAR Protected	26	LOGSTREAM_NAME	logstream name
(222)	CHAR Protected	2	*	reserved
(224)	CHAR Protected	28	LOGSTREAM_STATS	statistics
(224)	SIGNED Protected	4	FORCE_WAITS_CU	current, peak and
(228)	SIGNED Protected	4	FORCE_WAITS_PK	total waiters for
(22C)	SIGNED Protected	4	FORCE_WAITS_TO	Current buffer force
(230)	SIGNED Protected	4	BUF_FULL_WAITS	total waiters for Previous buffer write
(234)	SIGNED Protected	4	BUF_APPENDS	No of buffer appends
(238)	CHAR Protected	8	*	reserved for stats
(240)	UNSIGNED Protected	4	*	Deferred force

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(240)	FIXED Protected	1	DEFER_FORCE_ FLAG	active flag. 31 bits resvd.
(244)	CHAR Protected	4	*	
(248)	CHAR Protected	24	LOGSTREAM_ OPT_FIELDS	Wait optimiser
(248)	CHAR Protected	6	*	Reserved
(24E)	CHAR Protected	8	INTERVAL_START	STCK of start
(24E)	UNSIGNED Protected	2	START_HIGH	High order hword
(250)	UNSIGNED Protected	4	START_TIME	16 microsecond units
(254)	CHAR Protected	2	*	Reserved
(256)	CHAR Protected	2	*	Reserved
(258)	SIGNED Protected	4	LAST_FORCE_ TASK	Last forcing tsk
(25C)	SIGNED Protected	4	AVERAGE_GAP	Average gap
(260)	CHAR Protected		*	round to double word

--
-

The Stream class data consists of

- An eyecatcher.

- The anchor of a doubly-linked list of all Streams.

- An object factory instance used to allocate Stream objects.

- The current value of the deferred flush interval.

SHARED DATA

Declared Data				
(0)	CHAR Protected	128	CLASSDATABLOCK	
(0)	STRUCTURE IsA(L2_EYE_CATCHER) Protected	16	CLASS_EYE_ CATCHER	an eye-catcher
(0)	UNSIGNED Public	2	L2_EYE_LEN	object length
(2)	UNSIGNED Public	2	L2_EYE_OFFSET	offset of eye-catcher in object
(4)	CHAR Public	12	L2_EYE_STRING	'>DFHL2xxxxxx'
(10)	OBJECT IsA(HOP_DCHAIN) Protected	40	GLOBAL_ STREAM_CHAIN	chain of Streams
Inherited Data				
(10)	CHAR Private	4	*	
(18)	CHAR Protected	16	ITER0	
(18)	CHAR Private	4	*	
(20)	CHAR Protected	8	*	
(20)	ADDRESS Protected	4	PREV	
(24)	ADDRESS Protected	4	NEXT	
(28)	CHAR Protected	16	NODE0	
(28)	CHAR Private	4	*	
(30)	CHAR Protected	8	*	
(30)	ADDRESS Protected	4	PREV	
(34)	ADDRESS Protected	4	NEXT	
(38)	OBJECT IsA(L2OF) Protected	40	STREAM_FACTORY	Stream factory

-

The instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix 'L2OF' and a suffix which is the name of the object being managed.

(38)	CHAR Protected	40	INSTANCE_ DATA_BLOCK	L2OF instance data
(38)	CHAR Protected	16	OF_EYE_ CATCHER	eye-catcher
(38)	UNSIGNED Public	2	L2_EYE_LEN	object length
(3A)	UNSIGNED Public	2	L2_EYE_ OFFSET	offset of eye-catcher in object
(3C)	CHAR Public	12	L2_EYE_ STRING	'>DFHL2xxxxxx'
(48)	CHAR Protected	8	SUBPOOL_NAME	subpool name

Offset Hex	Type	Len	Name (Dim)	Description
(48)	CHAR Protected	4	SUBPOOL_ NAME_PREFIX	subpool name prefix
(4C)	CHAR Protected	4	SUBPOOL_ NAME_SUFFIX	subpool name suffix
(50)	CHAR Protected	8	SUBPOOL_TOKEN	subpool token
(58)	CHAR Protected	8	*	
(60)	UNSIGNED Protected	4	DEFER_FORCE_ INTERVAL	Current value
(64)	CHAR Protected	28	*	reserved

--
-

Declare Stream associated types. There is a type for the token by which a Stream may be referred to, for the Stream view of a Block, for the state that this view may be in, and for an element used to identify a task that suspends while writing to Current or forcing Current or Previous.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	ADDRESS Public	4	LOGSTREAMTOKEN	
(0)	FIXED Protected	1	BLOCKSTATUS	

-

Stream has its own view of a Block and the state it is in. Each Stream object contains two of these. At any given time, one will be for Current and the other will be for Previous. Each such StreamBlock contains:

- A pointer to the actual corresponding Block object.

- The current state of the block, which is used to manage the deferred force, write and wait protocols.

- The ForceToken associated with the block. Stream also uses this to uniquely identify the block. It will be zero if no records have yet been appended.

- When the block is in +flushed+ state, the pointer to the new Block object to be used as the new Current when the next buffer switch occurs.

- The nominal owner of the block. This is set when deferring the force of the Current block or waiting for the Previous block to harden, and is the transaction number of the task performing the action. It is only for debugging purposes.

- A queue of tasks which are suspended waiting for a force or write to complete for the block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	64	STREAMBLOCK	
(0)	ADDRESS Protected	4	BLOCK_PTR	-> actual Block object
(4)	UNSIGNED Protected	4	FORCE_TOKEN	force token for block
(8)	ADDRESS Protected	4	NEXT_BLOCK_PTR	-> next Block to be Current
(C)	CHAR Protected	4	BLOCK_OWNER	tran number of nominal owner
(10)	OBJECT IsA(HOP_DCHAIN) Protected	40	SUSPEND_QUEUE	chain of suspended tasks
(10)	CHAR Private	4	*	
(18)	CHAR Protected	16	ITER0	
(18)	CHAR Private	4	*	
(20)	CHAR Protected	8	*	
(20)	ADDRESS Protected	4	PREV	
(24)	ADDRESS Protected	4	NEXT	
(28)	CHAR Protected	16	NODE0	
(28)	CHAR Private	4	*	
(30)	CHAR Protected	8	*	
(30)	ADDRESS Protected	4	PREV	
(34)	ADDRESS Protected	4	NEXT	
(38)	FIXED Protected	1	STATUS	current status
(39)	CHAR Protected	7	*	

Offset Hex	Type	Len	Name (Dim)	Description

--
-

Declare a type for an element used to identify a task that suspends while writing to Current or forcing Current or Previous. This contains:

- A link to other such elements in a chain.
- The default suspend token for the task.
- The owner of this element, ie, the suspending task.
- The state of the associated block when the task suspended.
- The ForceToken of the associated block when the task suspended, enabling the task to see if the block was forced while it was suspended.
- A flag indicating if the task has been resumed and dechained, used to implement Dispatcher protocol.

(0)	CHAR Protected	32	SUSPENDELEMENT	
(0)	OBJECT IsA(HOP_DCHAINNODE) Protected	16	LINK	chain link
(0)	CHAR Private	4	*	
(8)	CHAR Protected	8	*	
(8)	ADDRESS Protected	4	PREV	
(C)	ADDRESS Protected	4	NEXT	
(10)	ADDRESS Protected	4	SUSPEND_TOKEN	suspend token
(14)	CHAR Protected	4	OWNER	tran number of task
(18)	UNSIGNED Protected	4	SUSPEND_FTOKEN	force token of block
(1C)	FIXED Protected	1	SUSPEND_STATUS	state when suspended
(1D)	FIXED Protected	1	DECHAINED	dechained and resumed?
(1E)	CHAR Protected	2	*	

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	RESET	
1	DECIMAL	2	DEFERRAL_ACTIVE	
1	DECIMAL	3	DEFERRAL_OVER	
1	DECIMAL	4	START_WRITE_ISSUED	
1	DECIMAL	5	START_WRITE_ COMPLETE	
1	DECIMAL	6	WAIT_WRITE_ISSUED	
1	DECIMAL	7	FLUSHED	
4	CHARACTER	AL2C	L2SR_LOCK_ ERROR_CODE	
4	CHARACTER	AL2D	L2SR_UNLOCK_ ERROR_CODE	
2	CHARACTER		NO_SOURCE	
8	CHARACTER		NO_JOURNAL	
4	DECIMAL	1	BUFFER_FULL	
4	DECIMAL	2	AKP_KICK_OFF	
4	DECIMAL	4	BUFFER_LENGTH_ ERROR	
4	DECIMAL	8	LOST_DATA	
4	DECIMAL	9	LOST_ACCESS	
4	DECIMAL	3	DATA_NOT_FOUND	
4	DECIMAL	5	END_OF_DATA	
4	DECIMAL	11	OUT_OF_RANGE	
4	DECIMAL	10	EMPTY_STREAM	
4	DECIMAL	21	RETRY_APPEND	
4	DECIMAL	6	CONNECT_FAILURE	
4	DECIMAL	7	LOG_NOT_DEFINED	
4	DECIMAL	20	NOT_POSSIBLE	

MEMMS Message table definition

```
  MODULE NAME = DFHMEMMS COPY
  DESCRIPTIVE NAME = CICS MESSAGE DOMAIN - STRUCTURE OF DATA
                 IN MESSAGE DEFINITION MODULE (DFHMET)
                       to be generated
 FUNCTION= This member describes the structure of data contained
      in the Message Definition Table (DFHMET). It provides
      symbolic access to the message templates, together
      with the globals in storage created by message domain
      initialisation.
      (a) The Message Domain (DFHMEx)
      (b) Message Module CMS Utility (DFHMEU) to build
        the message module from CMS Source data in DFHMET.
   Module Header
 FUNCTION= This member describes the structure of data contained
      in the Message Definition Table (DFHMET). It provides
      symbolic access to the message templates, together
      with the globals in storage created by message domain
      initialisation.
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	36	MET_MODULE_HEADER	
(0)	UNSIGNED	1	MET_HEADER_ LENGTH	length of header data
(1)	CHARACTER	1	METH_ARROW	Arrow '>'
(2)	CHARACTER	8	METH_MODULE_ IDENT	Module name
(A)	CHARACTER	4	METH_RELEASE	Product release code
(E)	CHARACTER	8	METH_PTFLEVEL	Service PTF level
(16)	CHARACTER	8	METH_ASMDATE	Assembly date mm/dd/yy
(1E)	CHARACTER	1	METH_AT_SYMBOL	
(1F)	CHARACTER	5	METH_ASMTIME	Assembly time hh.mm

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METX_MESSAGE_ INDEX	
(0)	UNSIGNED	2	METX_INDEX_ LENGTH	length of index data
(2)	CHARACTER	3	METX_MESSAGE_ PREFIX	Prefix e.g. DFH
(5)	UNSIGNED	1	METX_INDEX_ ENTRIES	No.of index entries
(6)	UNSIGNED	1	METX_ENTRY1_ OFFSET	Offset of 1st entry
(7)	CHARACTER	1	*	Padding for alignment
(8)	CHARACTER	*	METX_INDEX_DATA	Start of index data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	METX_INDEX_ENTRY	Generalised indexentry
(0)	CHARACTER	2	METX_MSGSET_NAME	Message set name (nn)
(2)	CHARACTER	2	*	Padding (for aligned V-con to follow)
(4)	ADDRESS	4	METX_MSGSET_ ADDRESS	Address of start of these messages

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	96	METG_MESSAGE_ GLOBALS	
(0)	UNSIGNED	2	METG_AREA_LENGTH	length of globals data
(2)	CHARACTER	10	METG_DATE_FORMAT	e.g. dd-mm-yyyy
(C)	CHARACTER	9	METG_TIME_FORMAT	e.g.hh-mm-ssX, where (X denotes am/pm form)
(15)	CHARACTER	3	METG_NEGNO_ FORMAT	e.g. -n or (n)
(18)	CHARACTER	7	METG_DECIMAL_ FORMAT	e.g. 1,234.5
(1F)	CHARACTER	10	METG_NUMERIC_SET	e.g. 0123456789
(29)	CHARACTER	1	METG_REPLY_FOLD	'Y'=fold 'N'=nofold
(2A)	CHARACTER	54	*	(Reserved)

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	36	METM_HEADER	
(0)	UNSIGNED	1	METM_HEADER_ LENGTH	length of header data (includes this field) *
(1)	CHARACTER	1	METM_ARROW	Arrow '>'
(2)	CHARACTER	8	METM_MODULE_ IDENT	Module name
(A)	CHARACTER	4	METM_RELEASE	Product release code *
(E)	CHARACTER	8	METM_PTFLEVEL	Service PTF level
(16)	CHARACTER	8	METM_ASMDATE	Assembly date mm/dd/yy *

Offset Hex	Type	Len	Name (Dim)	Description
(1E)	CHARACTER	1	METM_AT_SYMBOL	
(1F)	CHARACTER	5	METM_ASMTIME	Assembly time hh.mm *

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1	METM_MESSAGE_ COMPONENT	
(0)	UNSIGNED	1	METM_MSG_ COMPONENT_TYPE	component type Constant values of METM_MSG_COMPONENT_TYPE

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	METM_MESSAGE_DEFN	
(0)	CHARACTER	1	*	component identifier
(1)	UNSIGNED	2	METM_MSGDEF_ LENGTH	length of message definition
(3)	UNSIGNED	2	METM_MSGENTRY_ LENGTH	length of entire entry including symstring def
(5)	UNSIGNED	2	METM_USER_ EXIT_OFFSET	Offset of User exit data from start of msg *
(7)	BITSTRING	1	*	
	1...		METM_SYMSTRING	Flag set if message has
	.111 1111		*	symstring def
(8)	FULLWORD	4	METM_SPECINS_ INDICATOR	Reserved for special * insert indicators
(8)	UNSIGNED	1	METM_SPECINS_ GEN	
	1...		METM_DATE	Date
	.1..		METM_TIME	Time
	..1.		METM_APPLID	Applid
	...1		METM_SYSID	Sysid
	 1111		*	Reserved
(9)	UNSIGNED	1	METM_SPECINS_TM	
	1...		METM_TRANID	Tranid
	.1..		METM_TERMID	Termid
	..1.		METM_USERID	userid
	...1		METM_NETNAME	netname
	.... 1...		METM_TRANNUM	Transaction num
	.... .111		*	Reserved
(A)	UNSIGNED	1	METM_SPECINS_PC	
	1...		METM_PROGNAME	Program name
	.1..		METM_PRIMAB	Primary abcode
	..1.		METM_SECAB	Secondary abcode
	...1 1111		*	Reserved

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	11	METM_MESSAGE_ IDENT	
(0)	CHARACTER	1	*	component identifier
(1)	UNSIGNED	1	METM_MSGIDENT_ LENGTH	component length
(2)	CHARACTER	2	METM_COMPONENT_ ID	CICS domain(component)
(4)	UNSIGNED	2	METM_MESSAGE_NO	halfword message no.
(6)	CHARACTER	2	METM_MESSAGE_ CODES	
(6)	CHARACTER	1	METM_OPERATOR_ ACTION	operator action code
(7)	CHARACTER	1	METM_SEVERITY	severity code
(8)	UNSIGNED	2	METM_RESP2_VALUE	halfword EIBRESP2
(A)	CHARACTER	1	METM_NOREROUTE	noreroute flg

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	3	METM_MSG_ DESTINATIONS	
(0)	CHARACTER	1	*	component identifier
(1)	UNSIGNED	1	METM_MSGDESTS_ LENGTH	component length
(2)	UNSIGNED	1	METM_DEST_TYPES	dest types
	1...		METM_CONSOLE	type console
	.1..		METM_TDQ	type tdq
	..1.		METM_TERMENDU	type terminal end user
	...1		METM_TERMCDBC	type terminal CDBC *
	.... 1...		METM_SYSPRINT	SYSPRINT

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_MSG_RCS	
(0)	CHARACTER	1	*	component identifier
(1)	UNSIGNED	1	METM_RC_ELEMS	number of route codes
(2)	UNSIGNED	1	METM_RC_DATA (*)	list of 1 byte route code *

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_MSG_TDQS	
(0)	CHARACTER	1	*	component identifier
(1)	UNSIGNED	1	METM_TDQ_ELEMS	number of TDQs
(2)	CHARACTER	4	METM_TDQ_DATA (*)	list of TDQs each 4 bytes *

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_MSG_TEMPLATE	
(0)	CHARACTER	1	*	component identifier
(1)	UNSIGNED	1	METM_TEMPLATE_ ELEMS	no.of template elemnts
(2)	CHARACTER	*	METM_TEMPLATE_ DATA	template data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_ELEMENT	
(0)	UNSIGNED	1	METM_ELEMENT_ TYPE	element code
(1)	CHARACTER	*	METM_ELEM_DATA	Constant values of METM_ELEMENT_TYPE

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_TEXT_ELEMENT	
(0)	CHARACTER	1	*	element code
(1)	UNSIGNED	1	METM_TEXT_ EL_LENGTH	text string length
(2)	CHARACTER	*	METM_TEXT_STRING	text string

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_INSERT_ ELEMENT	
(0)	CHARACTER	1	*	element code
(1)	UNSIGNED	1	METM_INSERT_ID	insert identifier no
(2)	UNSIGNED	1	METM_INSERT_ FORMAT	insert format
(3)	CHARACTER	*	METM_OPTVALUES_ DATA	optional values data
(3)	UNSIGNED	1	METM_OPTVALUES_ COUNT	no.of optional values Constant values of METM_INSERT_FORMAT

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_OPTIONAL_ INSERT	
(0)	UNSIGNED	1	METM_OPTINS_ IDENT	option value number
(1)	UNSIGNED	1	METM_OPTINS_ LENGTH	value text length
(2)	CHARACTER	*	METM_OPTINS_TEXT	value text string

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_REPLY_ ELEMENT	
(0)	CHARACTER	1	*	element code
(1)	UNSIGNED	1	METM_REPLY_IDENT	reply value number
(2)	UNSIGNED	1	METM_REPLY_ LENGTH	reply text length
(3)	CHARACTER	*	METM_REPLY_TEXT	reply text string

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_SPECIAL_ INSERT_ELEMENT	
(0)	CHARACTER	1	*	element code
(1)	UNSIGNED	1	METM_SPECIAL_ INSERT_ELEMS	No of special inserts *

Offset Hex	Type	Len	Name (Dim)	Description
(2)	UNSIGNED	1	METM_SPECIAL_ INSERT_FORMAT (*)	special insert * type values

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_EXIT_MAP	
(0)	CHARACTER	1	*	Component identifier
(1)	UNSIGNED	1	METM_EXIT_ELEMS	no of exit elements
(2)	CHARACTER	2	METM_EXIT_DATA (*)	array of exit data
(2)	UNSIGNED	1	METM_EXIT_TYPE	either ins# or special *
(3)	UNSIGNED	1	METM_EXIT_ FORMAT	type code of insert

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_SYMSTRING_ DEFINITION	
(0)	CHARACTER	1	*	comp identifier
(1)	UNSIGNED	1	METM_SYMPTOM_ ELEMS	no. of extra symps
(2)	CHARACTER	*	METM_SYMSTRING_ DEFINITION_DATA	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_SYMPTOM	
(0)	UNSIGNED	1	METM_SYMPTOM_ TYPE	
(1)	UNSIGNED	1	METM_SYMPTOM_ DATA_TYPE	
(2)	CHARACTER	*	METM_SYMPTOM_ DATA	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	METM_SYMPTOM_ INSERT_DATA	
(0)	CHARACTER	1	*	Symptom type
(1)	CHARACTER	1	*	Symptom data type
(2)	UNSIGNED	2	METM_SYMPTOM_ INSERT_OFFSET	from msgdef start

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	3	METM_SYMPTOM_ SPECIAL_DATA	
(0)	CHARACTER	1	*	Symptom type
(1)	CHARACTER	1	*	Symptom data type
(2)	UNSIGNED	1	METM_SYMPTOM_ SPECIAL_TYPE	special-insert type declared above

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	METM_SYMPTOM_ TEXT_DATA	
(0)	CHARACTER	1	*	Symptom type
(1)	CHARACTER	1	*	Symptom data type
(2)	UNSIGNED	1	METM_SYMPTOM_ TEXT_LENGTH	Length of string
(3)	CHARACTER	*	METM_SYMPTOM_ TEXT_STRING	

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	START_OF_MESSAGE	
1	DECIMAL	2	MESSAGE_IDENT	
1	DECIMAL	3	MESSAGE_DEST	
1	DECIMAL	4	MESSAGE_TEMPLATE	
1	DECIMAL	5	END_OF_MESSAGE	
1	DECIMAL	6	MESSAGE_TDQS	new TDQ list
1	DECIMAL	7	MESSAGE_RCS	new route code list
1	DECIMAL	8	SYMSTRING_DEF	
1	DECIMAL	9	END_OF_SYMSTRING	
1	DECIMAL	10	USER_EXIT_MAP	
1	DECIMAL	255	END_OF_MODULE	
4	DECIMAL	28	MAX_ROUTE_CODES	
4	DECIMAL	25	MAX_QUEUES	
1	DECIMAL	1	TEXT_ELEMENT	
1	DECIMAL	2	INSERT_ELEMENT	
1	DECIMAL	3	REPLY_ELEMENT	
1	DECIMAL	4	SPECIAL_INSERT_ ELEMENT	
1	DECIMAL	1	FORMAT_CHAR	
1	DECIMAL	2	FORMAT_HEX	
1	DECIMAL	3	FORMAT_DEC	
1	DECIMAL	4	FORMAT_OPT	
1	DECIMAL	5	FORMAT_DATE	
1	DECIMAL	6	FORMAT_TIME	

Constant values used to represent inserts/special-inserts/symptom arg

1	DECIMAL	1	INSERT1	
1	DECIMAL	2	INSERT2	
1	DECIMAL	3	INSERT3	
1	DECIMAL	4	INSERT4	
1	DECIMAL	5	INSERT5	
1	DECIMAL	6	INSERT6	
1	DECIMAL	7	INSERT7	
1	DECIMAL	8	INSERT8	
1	DECIMAL	9	INSERT9	
1	DECIMAL	10	INSERT10	
1	DECIMAL	11	SPECIAL_TIME	
1	DECIMAL	12	SPECIAL_DATE	
1	DECIMAL	13	SPECIAL_APPLID	
1	DECIMAL	14	SPECIAL_SYSID	
1	DECIMAL	15	SPECIAL_TRANID	
1	DECIMAL	16	SPECIAL_TERMID	
1	DECIMAL	17	SPECIAL_PROGNAME	
1	DECIMAL	18	SPECIAL_USERID	
1	DECIMAL	19	SPECIAL_NETNAME	
1	DECIMAL	20	SPECIAL_TRANNUM	
1	DECIMAL	21	SPECIAL_PRIMAB	
1	DECIMAL	22	SPECIAL_SECAB	

This further member is needed as common code is shared with the symptom string code. Apart from the above text strings are allowable as symptom arguments.

4	DECIMAL	23	TEXT_STRING	
1	DECIMAL	1	SYMPTOM_INSERT	
1	DECIMAL	2	SYMPTOM_SPECIAL	
1	DECIMAL	3	SYMPTOM_TEXT	

MEPS Message domain anchor block

```
MODULE NAME = DFHMEPS COPY
DESCRIPTIVE NAME = CICS Message Domain Anchor Block
FUNCTION = This member describes the structure of the data
        contained in the ME domain Anchor Block. It also
        contains the global variables used throughout the
        ME domain, eg. YES, NO, ON, OFF, etc, the NLS Table,
        and the ME Catalog Record.
        The ME domain Anchor block is set up during
        Pre-initialise and Initialise, by DFHMEDM. It
        remains until CICS is terminated.
        The anchor block contains the necessary system
        options for Messages, eg the Languages in the system
        and the default language etc. These are SIT options,
        and are assumed to hold true until the next CICS
        start.
      1) DFHSUME
      2) DFHMEDM
      3) DFHMEBU
      4) DFHMEIN
      5) DFHMESR
      6) DFHMEME
      7) DFHMEDUF
      8) DFHMEWS
                     be generated
                     big enough when DFHMET19 missing
 ME domain common structures and constants
 ME Anchor Block
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	577	ANCHOR	Anchor block
(0)	CHARACTER	16	ANCH_PREFIX	Anchor prefix area
(0)	HALFWORD	2	ANCH_LENGTH	Anchor length
(2)	CHARACTER	1	ANCH_ARROW	Arrow eyecatcher
(3)	CHARACTER	3	ANCH_DFH	DFH
(6)	CHARACTER	2	ANCH_DOMID	Domain id
(8)	CHARACTER	8	ANCH_BLOCK_NAME	Control block name
(10)	CHARACTER	4	LOCKING_INFO	ME Locking Information
(10)	ADDRESS	4	LOCK_TOKEN	ME Lock Token
(14)	BITSTRING	1	PHASE_INFO	Phase information
	1...		PRE_INIT_ COMPLETE_FLAG	Pre-initialise complete
	.1..		XMEOUT_ACTIVE	User exit active flag
	..11 1111		*	Reserved
(15)	BITSTRING	1	RECOVERY_INFO	Recovery information
(15)	BITSTRING	1	*	Reserved
(16)	UNSIGNED	1	MESSAGE_CASE	Message case required
(17)	CHARACTER	1	*	Reserved
(18)	CHARACTER	552	MESSAGE_INFO	Message Information
(18)	CHARACTER	1	DEFAULT_ LANGUAGE	One-character default language suffix
(19)	CHARACTER	3	DEFAULT_ LANGUAGE_CODE	Three-letter default language code
(1C)	UNSIGNED	1	NUMBER_OF_LANGS	Number in this system
(1D)	UNSIGNED	1	ME_DOMAIN_ STATUS	Status flag
(1E)	CHARACTER	2	*	Reserved
(20)	ADDRESS	4	NLS_TABLE_PTR	Pointer to NLS Table
(24)	ADDRESS	4	DEFAULT_ LANG_PTR	Default language Ptr
(28)	CHARACTER	36	LANGUAGES_USED	Languages available in the system
(4C)	CHARACTER	57	UNAVAILABLE_ LANGUAGES	Languages noted as not available in the system
(85)	CHARACTER	3	*	Reserved
(88)	ADDRESS	4	MSG_MOD_PTRS (36)	Array of ptrs, one for each message module
(118)	ADDRESS	4	FEATURE_ DEFAULT_LANG_PTR	Default feature table pointer
(11C)	ADDRESS	4	USER_DEFAULT_ LANG_PTR	Default user table pointer
(120)	ADDRESS	4	FEATURE_ MSG_MOD_PTRS (36)	Array of ptrs, * one for each feature * message module
(1B0)	ADDRESS	4	USER_MSG_ MOD_PTRS (36)	Array of ptrs, one * for each user message * module
(240)	BITSTRING	1	MSG_LEVEL_INFO	Msg Level Information *
	1...		MSG_LEVEL	Message Level
	.111 1111		*	Reserved

National Language Support Table (NLS_TABLE).
NLS_TABLE consists of three-letter national language codes and
one-character CICS language suffixes.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	NLS_TABLE (54)	Each entry in NLS_TABLE consists of a
(0)	CHARACTER	3	NLS_CODE	three-letter language code,
(3)	CHARACTER	1	NLS_SUFFIX	and a one-character language suffix

ME Catalogue Record

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	44	CATALOG_RECORD	ME catalogue record
(0)	UNSIGNED	1	MECR_MESSAGE_ CASE	Message case required
(1)	UNSIGNED	1	MECR_NUMBER_ OF_LANGS	
				Number in this system
(2)	CHARACTER	36	MECR_LANGUAGES_ USED	
				Langs in system
(26)	CHARACTER	1	MECR_DEFAULT_ LANGUAGE	
				System default language System default language code
(27)	CHARACTER	3	MECR_DEFAULT_ LANGUAGE_CODE	
(2A)	CHARACTER	1	*	Reserved
(2B)	BITSTRING	1	*	
	1...		MECR_MSG_LEVEL	Message Level
	.111 1111		*	Reserved

Generalised insert structure - used as an overlay for the CDURUN.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	GENERAL_INSERT	INSERTn
(0)	ADDRESS	4	GEN_INSERT_PTR	-> INSERTn_P
(4)	FULLWORD	4	GEN_INSERT_LEN	INSERTn_N

Storage to build record into

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	532	SYMPTOM_RECORD	
(0)	CHARACTER	1	SYMPTOM_ RECORD_CHAR (532)	
				@D4

Constants

Len	Type	Value	Name	Description
1	CHARACTER	>	ARROW	> for prefix
4	DECIMAL	32	BDY32	Used for storage bdy
0	BIT	1	YES	Yes
0	BIT	0	NO	No
0	BIT	1	ON	On
0	BIT	0	OFF	Off
1	DECIMAL	1	UPPER	upper case messages
1	DECIMAL	2	MIXED	mixed case messages
4	DECIMAL	4	POINT_ID_LENGTH	Length of point_id
1	DECIMAL	1	BIT_ON	Represents a bit set on
1	DECIMAL	0	BIT_OFF	Represents a bit set off
1	DECIMAL	1	ZSUPP_YES	Suppress leading 0's
1	DECIMAL	0	ZSUPP_NO	Don't suppress leading 0's
4	DECIMAL	196	MAX_SYMPTOM_STRING_LEN	Max length of a symptom string
1	HEX	00	NULL_LANGUAGE	Null language suffix
Message Domain Status Constants				
4	DECIMAL	1	PRE_INITIALISED	
4	DECIMAL	2	INITIALISED	
4	DECIMAL	3	QUIESCING	
4	DECIMAL	4	TERMINATING	
Maximum Values Constants				
1	DECIMAL	36	MAX_LANGUAGES	Maximum Number of languages allowed in the system *
1	DECIMAL	20	MAX_REPLIES	Maximum number of replies allowed in a message *
1	DECIMAL	10	MAX_INSERTS	Maximum number of inserts allowed in a message * Number of supported three-letter language codes in NLS_TABLE
2	DECIMAL	54	NUMBER_OF_LANGUAGE_CODES	
Symptom Record				
4	DECIMAL	312	SR_FIXED_STORAGE	@D4
4	DECIMAL	220	SR_PRIMLEN	@D4
4	DECIMAL	0	SR_SECLEN	Not using secondary @D4
4	DECIMAL	0	SR_VARLEN	Not using variable @D4
4	DECIMAL	532	SR_TOTAL_LEN	@D4

MNAFB Monitoring authorised parameter block

CONTROL BLOCK NAME = DFHMNAFB
DESCRIPTIVE NAME = CICS/MVS Monitoring (MN) Domain
    Authorised Facilities Parameter Block
Function =
  This file contains the control block and constant
  declarations for the parameter list used by Monitoring
  for communication between the functional gate and the
  SVC service routine.
LIFETIME =
STORAGE CLASS = N/A
LOCATION =
INNER CONTROL BLOCKS = None
Notes:
 Dependencies = S/370
 Restrictions = None
 Register Conventions = Domain standard (no special usage)
 Patch Label = N/A
 Module Type = Control block definition
 Attributes = N/A
EXTERNAL REFERENCES = None
  DATA AREAS = None
  CONTROL BLOCKS = None
  GLOBAL VARIABLES (Macro pass) = None
Monitor Authorised Facilities Parm Block -- M A F P B --
The Monitor Authorised Facilities Parameter Block contains:
    The authorised facility function code.
    The function return code.
    The SMF record address
    The SYSEVENT record address
    The MVS Workload Manager fields
    The creation time of the MAFPB

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	68	MAFPB	
	Prefix fields for restructured control blocks			
(0)	CHARACTER	16	MAFPB_PREFIX	
(0)	UNSIGNED	2	MAFPB_LENGTH	
(2)	CHARACTER	1	MAFPB_ARROW	
(3)	CHARACTER	3	MAFPB_DFH	
(6)	CHARACTER	2	MAFPB_DOMAIN	
(8)	CHARACTER	8	MAFPB_BLOCK_ID	
	Function the Monitoring authorised module should perform, ie SMF write, or MVS SRM notify			
(10)	UNSIGNED	2	MAFPB_FUNCTION	
	Monitoring authorised module return code. It is not the SMF or SYSEVENT return code. If this is set to MAFPB_ SYSEVENT_ ERROR or MAFPB_SMF_ERROR, the respective return code is in MAFPB_ SYSEVENT_ RC or MAFPB_SMF_RC.			
(12)	UNSIGNED	1	MAFPB_RESPONSE	
	Indicator to Monitoring authorised module whether to perform GTF tracing.			
(13)	BITSTRING	1	*	
	1...		MAFPB_GTF_TRACE_FLAG	
	.111 1111		*	
	Address of SMF record if SMF write is required.			
(14)	ADDRESS	4	MAFPB_SMF_RECORD	
	Addrress of SYSEVENT record if MVS SRM notification is required.			
(18)	ADDRESS	4	MAFPB_SYSEVENT_RECORD	
	SMF and SYSEVENT return codes			
(1C)	UNSIGNED	1	MAFPB_SMF_RC	
(1D)	UNSIGNED	1	MAFPB_SYSEVENT_ RC	
(1E)	UNSIGNED	2	*	
	MVS Return Code registers after SYSEVENT or SMFEWTM macros have been issued.			
(20)	FULLWORD	4	MAFPB_RTNREG0	
(24)	FULLWORD	4	MAFPB_RTNREG1	
(28)	FULLWORD	4	MAFPB_RTNREG15	

Offset Hex	Type	Len	Name (Dim)	Description
	MVS Workload Manager Connect Token, Performance Block Token, and the Transaction End Time for Report or Notify.			
(2C)	UNSIGNED	4	MAFPB_WLM_CONNECT_TOKEN	
(30)	UNSIGNED	4	MAFPB_WLM_PERFORMANCE_BLOCK	
(34)	BITSTRING	8	MAFPB_WLM_TRAN_END_TIME	
	STCK timestamp of either the MAFPB creation time, or the last time a record was written to SMF, or passed to the MVS SRM (SYSEVENT).			
(3C)	CHARACTER	8	MAFPB_CREATION_ STCK	
(44)	CHARACTER		*	

Constants

Len	Type	Value	Name	Description
2	DECIMAL	1	MAFPB_SMFEWTM	
2	DECIMAL	2	MAFPB_SYSEVENT	
2	DECIMAL	3	MAFPB_WLM_CONNECT	
2	DECIMAL	4	MAFPB_WLM_ DISCONNECT	
2	DECIMAL	5	MAFPB_WLM_REPORT	
2	DECIMAL	6	MAFPB_WLM_NOTIFY	
2	DECIMAL	7	MAFPB_WLM_ PB_CREATE	
2	DECIMAL	8	MAFPB_WLM_ PB_DELETE	
	Trace flags			
0	BIT	1	MAFPB_GTF_TRACE_ON	
0	BIT	0	MAFPB_GTF_ TRACE_OFF	
	Response codes			
1	DECIMAL	0	MAFPB_OK	
1	DECIMAL	1	MAFPB_NO_FESTAE	
1	DECIMAL	2	MAFPB_NO_ STORAGE_253	
1	DECIMAL	3	MAFPB_NO_AUTHORISATION	
1	DECIMAL	4	MAFPB_NO_ STORAGE_SMF	
1	DECIMAL	5	MAFPB_INVALID_RECORD_LENGTH	
1	DECIMAL	6	MAFPB_NOT_CICS_RECORD	
1	DECIMAL	7	MAFPB_SMF_ERROR	
1	DECIMAL	8	MAFPB_SYSEVENT_ ERROR	
1	DECIMAL	9	MAFPB_WLM_CONNECT_FAILED	
1	DECIMAL	10	MAFPB_WLM_DISCONNECT_FAILED	
1	DECIMAL	11	MAFPB_WLM_REPORT_FAILED	
1	DECIMAL	12	MAFPB_WLM_NOTIFY_FAILED	
1	DECIMAL	13	MAFPB_WLM_PB_CREATE_FAILED	
1	DECIMAL	14	MAFPB_WLM_PB_DELETE_FAILED	
1	DECIMAL	15	MAFPB_NO_STORAGE_MNACB	
1	DECIMAL	16	MAFPB_NO_STORAGE_HASH	
1	DECIMAL	17	MAFPB_NO_STORAGE_HASH_ELEM	
1	DECIMAL	18	MAFPB_INVALID_ PB_TOKEN	
1	DECIMAL	19	MAFPB_WLM_OP_OUT_OF_SEQUENCE	
1	DECIMAL	254	MAFPB_INVALID_ FUNCTION	*
	Control Block eyecatcher string			
8	CHARACTER	MAFPB	MAFPB_ID_STRING	

MNCBS Monitoring domain control blocks

```
CONTROL BLOCK NAME = DFHMNCBS
DESCRIPTIVE NAME = CICS/MVS Monitoring (MN) Domain
               Control Block declarations.
Function =
    This file contains the control block and constant
    declarations used by the Monitoring domain.
    The file is included by each Monitoring domain module.
    The control blocks are:
       TMA - Transaction Monitoring Area.
       GLOBAL - Monitoring global storage area.
            - Dictionary Entry.
            - Connector Arrays.
       DUMP - Dump control values.
       MSGS - Message Numbers.
       TRACE - Trace point definitions.
    Each control block declaration is followed by the
    constant declarations related to it.
Notes:
 Dependencies = S/370
 Restrictions = none
 Register Conventions = domain standard (no special usage)
 Patch Label = N/A
 Module Type = N/A
 Attributes = N/A
EXTERNAL REFERENCES = None
  DATA AREAS = None
  CONTROL BLOCKS = None
  GLOBAL VARIABLES (Macro pass) = None
The MN Domain Transaction Monitoring Area (TMA)
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1504	TRANSACTION_ MONITORING_AREA	
	Prefix fields for restructured control blocks			
(0)	CHARACTER	16	TMA_PREFIX	
(0)	UNSIGNED	2	TMA_LENGTH	
(2)	CHARACTER	1	TMA_ARROW	
(3)	CHARACTER	3	TMA_DFH	
(6)	CHARACTER	2	TMA_DOMAIN	
(8)	CHARACTER	8	TMA_BLOCK_ID	
	Date and time of TMA creation.			
(10)	CHARACTER	8	TMA_CREATION_ STCK	
	Reserved fields			
(18)	CHARACTER	8	TMA_RESERVED_1	
(20)	ADDRESS	4	TMA_PARENT_TMA	
(24)	ADDRESS	4	TMA_CHILD_TMA	
(28)	UNSIGNED	4	TMA_DEPTH_COUNT	
(2C)	CHARACTER	4	TMA_RESERVED_2	
(30)	ADDRESS	4	TMA_USER_ AREA_PTR	
(34)	ADDRESS	4	TMA_DS_TOKEN	
(38)	CHARACTER	4	TMA_WLM_ SRC_TOKEN	
(3C)	CHARACTER	4	TMA_RESERVED_3	
	Pointer to the Monitoring anchor			
(40)	ADDRESS	4	TMA_MNA_PTR	
(44)	CHARACTER	1	TMA_CLASS_STATUS	
	1...		TMA_EXCEPTION_ STATUS	
	.1..		TMA_PERFORMANCE_ STATUS	
	..1.		TMA_SYSEVENT_ STATUS	
	...1 1111		*	
(45)	CHARACTER	3	*	
	Exception record count for this transaction.			
(48)	UNSIGNED	4	TMA_EXCEPTION_ COUNT	
(4C)	CHARACTER	4	*	
	Elapsed and CPU timing fields			
(50)	CHARACTER	8	TMA_ELAPSED_TIME	
(58)	CHARACTER	8	TMA_CPU_TIME	
(60)	CHARACTER	8	*	
(68)	CHARACTER	8	TMA_START_TIME	
	Last suspend (susptime) interval for I/O clocks			

Offset Hex	Type	Len	Name (Dim)	Description
(70)	CHARACTER	8	TMA_LAST_SUSPEND_INTERVAL	
Accumulated suspend deltas for composite clocks				
(78)	UNSIGNED	4	TMA_COMPOSITE_171_INTVL	
(7C)	UNSIGNED	4	TMA_COMPOSITE_171_INTVL_COUNT	
(80)	UNSIGNED	4	TMA_COMPOSITE_254_INTVL	
(84)	UNSIGNED	4	TMA_COMPOSITE_254_INTVL_COUNT	
(88)	CHARACTER	8	*	
Current values for high water mark calculations				
(90)	CHARACTER	52	TMA_CURRENT	
(90)	UNSIGNED	4	TMA_DFHSTOR_ 033_C	
(94)	UNSIGNED	4	TMA_DFHSTOR_ 106_C	
(98)	UNSIGNED	4	TMA_DFHSTOR_ 116_C	
(9C)	UNSIGNED	4	TMA_DFHSTOR_ 119_C	
(A0)	UNSIGNED	4	TMA_DFHSTOR_ 087_C	
(A4)	UNSIGNED	4	TMA_DFHSTOR_ 139_C	
(A8)	UNSIGNED	4	TMA_DFHSTOR_ 108_C	
(AC)	UNSIGNED	4	TMA_DFHSTOR_ 142_C	
(B0)	UNSIGNED	4	TMA_DFHSTOR_ 143_C	
(B4)	UNSIGNED	4	TMA_DFHSTOR_ 122_C	
(B8)	UNSIGNED	4	TMA_DFHSTOR_ 162_C	
(BC)	UNSIGNED	4	TMA_DFHSTOR_ 161_C	
(C0)	UNSIGNED	4	TMA_DFHSTOR_ 160_C	
Time of last storage change for occupancy calc.				
(C4)	CHARACTER	16	TMA_OCCUPANCY	
(C4)	UNSIGNED	4	TMA_DFHSTOR_ 095_O	
(C8)	UNSIGNED	4	TMA_DFHSTOR_ 107_O	
(CC)	UNSIGNED	4	TMA_DFHSTOR_ 118_O	
(D0)	UNSIGNED	4	TMA_DFHSTOR_ 121_O	
Depth of recursion counts for recursive clocks				
(D4)	CHARACTER	4	TMA_RECURSE_ COUNTS	
(D4)	UNSIGNED	4	TMA_DFHTASK_ 170_A	
Define CICS monitoring data fields				
(D8)	CHARACTER	376	TMA_BEGIN	
(D8)	CHARACTER	4	TMA_DFHTASK_001	
(DC)	CHARACTER	4	TMA_DFHTERM_002	
(E0)	CHARACTER	8	TMA_DFHCICS_089	
(E8)	CHARACTER	4	TMA_DFHTASK_004	
(EC)	CHARACTER	8	TMA_DFHCICS_005	
(F4)	CHARACTER	8	TMA_DFHCICS_006	
(FC)	CHARACTER	4	TMA_DFHTASK_031	
(100)	UNSIGNED	4	TMA_DFHTASK_109	
(104)	CHARACTER	8	TMA_DFHTASK_166	
(10C)	CHARACTER	8	TMA_DFHTERM_111	
(114)	CHARACTER	8	TMA_DFHPROG_071	
(11C)	CHARACTER	20	TMA_DFHTASK_097	
(130)	CHARACTER	8	TMA_DFHTASK_098	
(138)	CHARACTER	4	TMA_DFHCICS_130	
(13C)	UNSIGNED	4	TMA_DFHCICS_131	
(140)	CHARACTER	8	TMA_DFHTASK_132	
(148)	CHARACTER	8	TMA_DFHCICS_167	
(150)	CHARACTER	8	TMA_DFHCICS_168	
(158)	CHARACTER	4	TMA_DFHTASK_163	
(15C)	BITSTRING	8	TMA_DFHTASK_164	
(164)	UNSIGNED	4	TMA_DFHTERM_165	
(168)	CHARACTER	4	TMA_DFHTERM_169	
(16C)	CHARACTER	4	TMA_DFHTASK_124	
(170)	CHARACTER	16	TMA_DFHTASK_190	
(180)	CHARACTER	36	TMA_DFHCBTS_200	
(1A4)	CHARACTER	8	TMA_DFHCBTS_201	
(1AC)	CHARACTER	52	TMA_DFHCBTS_202	
(1E0)	CHARACTER	52	TMA_DFHCBTS_203	
(214)	CHARACTER	16	TMA_DFHCBTS_204	
(224)	CHARACTER	16	TMA_DFHSOCK_244	
(234)	CHARACTER	28	TMA_DFHTASK_082	
(250)	CHARACTER	496	TMA_RESET	
(250)	CHARACTER	4	TMA_DFHTASK_064	
(254)	CHARACTER	4	TMA_DFHPROG_113	
(258)	CHARACTER	4	TMA_DFHPROG_114	
(25C)	CHARACTER	4	TMA_DFHCICS_112	
(260)	UNSIGNED	4	TMA_DFHTERM_034	
(264)	UNSIGNED	4	TMA_DFHTERM_083	
(268)	UNSIGNED	4	TMA_DFHTERM_035	
(26C)	UNSIGNED	4	TMA_DFHTERM_084	
(270)	UNSIGNED	4	TMA_DFHTERM_067	
(274)	UNSIGNED	4	TMA_DFHTERM_085	
(278)	UNSIGNED	4	TMA_DFHTERM_068	

Offset Hex	Type	Len	Name (Dim)	Description
(27C)	UNSIGNED	4	TMA_DFHTERM_086	
(280)	UNSIGNED	4	TMA_DFHTERM_135	
(284)	UNSIGNED	4	TMA_DFHTERM_137	
(288)	UNSIGNED	4	TMA_DFHTERM_136	
(28C)	UNSIGNED	4	TMA_DFHTERM_138	
(290)	UNSIGNED	4	TMA_DFHTERM_069	
(294)	UNSIGNED	4	TMA_DFHSTOR_054	
(298)	UNSIGNED	4	TMA_DFHSTOR_105	
(29C)	UNSIGNED	4	TMA_DFHSTOR_117	
(2A0)	UNSIGNED	4	TMA_DFHSTOR_120	
(2A4)	UNSIGNED	4	TMA_DFHSTOR_033	
(2A8)	UNSIGNED	4	TMA_DFHSTOR_106	
(2AC)	UNSIGNED	4	TMA_DFHSTOR_116	
(2B0)	UNSIGNED	4	TMA_DFHSTOR_119	
(2B4)	CHARACTER	8	TMA_DFHSTOR_095	
(2B4)	UNSIGNED	4	*	
(2B8)	UNSIGNED	4	*	
(2BC)	CHARACTER	8	TMA_DFHSTOR_107	
(2BC)	UNSIGNED	4	*	
(2C0)	UNSIGNED	4	*	
(2C4)	CHARACTER	8	TMA_DFHSTOR_118	
(2C4)	UNSIGNED	4	*	
(2C8)	UNSIGNED	4	*	
(2CC)	CHARACTER	8	TMA_DFHSTOR_121	
(2CC)	UNSIGNED	4	*	
(2D0)	UNSIGNED	4	*	
(2D4)	UNSIGNED	4	TMA_DFHSTOR_144	
(2D8)	UNSIGNED	4	TMA_DFHSTOR_145	
(2DC)	UNSIGNED	4	TMA_DFHSTOR_146	
(2E0)	UNSIGNED	4	TMA_DFHSTOR_147	
(2E4)	UNSIGNED	4	TMA_DFHSTOR_148	
(2E8)	UNSIGNED	4	TMA_DFHSTOR_149	
(2EC)	UNSIGNED	4	TMA_DFHSTOR_087	
(2F0)	UNSIGNED	4	TMA_DFHSTOR_139	
(2F4)	UNSIGNED	4	TMA_DFHSTOR_108	
(2F8)	UNSIGNED	4	TMA_DFHSTOR_142	
(2FC)	UNSIGNED	4	TMA_DFHSTOR_143	
(300)	UNSIGNED	4	TMA_DFHSTOR_122	
(304)	UNSIGNED	4	TMA_DFHSTOR_162	
(308)	UNSIGNED	4	TMA_DFHSTOR_161	
(30C)	UNSIGNED	4	TMA_DFHSTOR_160	
(310)	UNSIGNED	4	TMA_DFHFILE_036	
(314)	UNSIGNED	4	TMA_DFHFILE_037	
(318)	UNSIGNED	4	TMA_DFHFILE_038	
(31C)	UNSIGNED	4	TMA_DFHFILE_039	
(320)	UNSIGNED	4	TMA_DFHFILE_040	
(324)	UNSIGNED	4	TMA_DFHFILE_093	
(328)	UNSIGNED	4	TMA_DFHFILE_070	
(32C)	UNSIGNED	4	TMA_DFHDEST_041	
(330)	UNSIGNED	4	TMA_DFHDEST_042	
(334)	UNSIGNED	4	TMA_DFHDEST_043	
(338)	UNSIGNED	4	TMA_DFHDEST_091	
(33C)	UNSIGNED	4	TMA_DFHTEMP_044	
(340)	UNSIGNED	4	TMA_DFHTEMP_046	
(344)	UNSIGNED	4	TMA_DFHTEMP_047	
(348)	UNSIGNED	4	TMA_DFHTEMP_092	
(34C)	UNSIGNED	4	TMA_DFHMAPP_050	
(350)	UNSIGNED	4	TMA_DFHMAPP_051	
(354)	UNSIGNED	4	TMA_DFHMAPP_052	
(358)	UNSIGNED	4	TMA_DFHMAPP_090	
(35C)	UNSIGNED	4	TMA_DFHPROG_055	
(360)	UNSIGNED	4	TMA_DFHPROG_056	
(364)	UNSIGNED	4	TMA_DFHPROG_057	
(368)	UNSIGNED	4	TMA_DFHPROG_072	
(36C)	UNSIGNED	4	TMA_DFHPROG_073	
(370)	UNSIGNED	4	TMA_DFHJOUR_058	
(374)	UNSIGNED	4	TMA_DFHJOUR_172	
(378)	UNSIGNED	4	TMA_DFHTASK_059	
(37C)	UNSIGNED	4	TMA_DFHTASK_066	
(380)	UNSIGNED	4	TMA_DFHSYNC_060	
(384)	UNSIGNED	4	TMA_DFHCICS_025	
(388)	UNSIGNED	4	TMA_DFHFEPI_150	
(38C)	UNSIGNED	4	TMA_DFHFEPI_151	
(390)	UNSIGNED	4	TMA_DFHFEPI_152	
(394)	UNSIGNED	4	TMA_DFHFEPI_153	
(398)	UNSIGNED	4	TMA_DFHFEPI_154	
(39C)	UNSIGNED	4	TMA_DFHFEPI_155	
(3A0)	UNSIGNED	4	TMA_DFHFEPI_157	
(3A4)	UNSIGNED	4	TMA_DFHFEPI_158	
(3A8)	UNSIGNED	4	TMA_DFHFEPI_159	
(3AC)	UNSIGNED	4	TMA_DFHCBTS_205	
(3B0)	UNSIGNED	4	TMA_DFHCBTS_206	
(3B4)	UNSIGNED	4	TMA_DFHCBTS_207	
(3B8)	UNSIGNED	4	TMA_DFHCBTS_208	
(3BC)	UNSIGNED	4	TMA_DFHCBTS_209	
(3C0)	UNSIGNED	4	TMA_DFHCBTS_210	

Offset Hex	Type	Len	Name (Dim)	Description
(3C4)	UNSIGNED	4	TMA_DFHCBTS_211	
(3C8)	UNSIGNED	4	TMA_DFHCBTS_212	
(3CC)	UNSIGNED	4	TMA_DFHCBTS_213	
(3D0)	UNSIGNED	4	TMA_DFHCBTS_214	
(3D4)	UNSIGNED	4	TMA_DFHCBTS_215	
(3D8)	UNSIGNED	4	TMA_DFHCBTS_216	
(3DC)	UNSIGNED	4	TMA_DFHCBTS_217	
(3E0)	UNSIGNED	4	TMA_DFHCBTS_218	
(3E4)	UNSIGNED	4	TMA_DFHCBTS_219	
(3E8)	UNSIGNED	4	TMA_DFHCBTS_220	
(3EC)	UNSIGNED	4	TMA_DFHCBTS_221	
(3F0)	UNSIGNED	4	TMA_DFHCBTS_222	
(3F4)	UNSIGNED	4	TMA_DFHWEBB_231	
(3F8)	UNSIGNED	4	TMA_DFHWEBB_232	
(3FC)	UNSIGNED	4	TMA_DFHWEBB_233	
(400)	UNSIGNED	4	TMA_DFHWEBB_234	
(404)	UNSIGNED	4	TMA_DFHWEBB_235	
(408)	UNSIGNED	4	TMA_DFHWEBB_236	
(40C)	UNSIGNED	4	TMA_DFHWEBB_237	
(410)	UNSIGNED	4	TMA_DFHDOCH_226	
(414)	UNSIGNED	4	TMA_DFHDOCH_227	
(418)	UNSIGNED	4	TMA_DFHDOCH_228	
(41C)	UNSIGNED	4	TMA_DFHDOCH_229	
(420)	UNSIGNED	4	TMA_DFHDOCH_230	
(424)	UNSIGNED	4	TMA_DFHDOCH_240	
(428)	UNSIGNED	4	TMA_DFHSOCK_242	
(42C)	UNSIGNED	4	TMA_DFHSOCK_243	
(430)	UNSIGNED	4	TMA_DFHDATA_179	
(434)	UNSIGNED	4	TMA_DFHDATA_180	
(438)	UNSIGNED	4	TMA_DFHTASK_248	
(43C)	UNSIGNED	4	TMA_DFHTASK_251	
(440)	CHARACTER	416	TMA_CLOCKS	
(440)	CHARACTER	8	TMA_DFHTASK_007	
(440)	UNSIGNED	4	TMA_DFHTASK_007_TIME	
(444)	BITSTRING	1	TMA_DFHTASK_007_FLAG	
(445)	UNSIGNED	3	TMA_DFHTASK_007_COUNT	
(448)	CHARACTER	8	TMA_DFHTASK_008	
(448)	UNSIGNED	4	TMA_DFHTASK_008_TIME	
(44C)	BITSTRING	1	TMA_DFHTASK_008_FLAG	
(44D)	UNSIGNED	3	TMA_DFHTASK_008_COUNT	
(450)	CHARACTER	8	TMA_DFHTASK_014	
(450)	UNSIGNED	4	TMA_DFHTASK_014_TIME	
(454)	BITSTRING	1	TMA_DFHTASK_014_FLAG	
(455)	UNSIGNED	3	TMA_DFHTASK_014_COUNT	
(458)	CHARACTER	8	TMA_DFHTASK_102	
(458)	UNSIGNED	4	TMA_DFHTASK_102_TIME	
(45C)	BITSTRING	1	TMA_DFHTASK_102_FLAG	
(45D)	UNSIGNED	3	TMA_DFHTASK_102_COUNT	
(460)	CHARACTER	8	TMA_DFHTASK_255	
(460)	UNSIGNED	4	TMA_DFHTASK_255_TIME	
(464)	BITSTRING	1	TMA_DFHTASK_255_FLAG	
(465)	UNSIGNED	3	TMA_DFHTASK_255_COUNT	
(468)	CHARACTER	8	TMA_DFHTASK_256	
(468)	UNSIGNED	4	TMA_DFHTASK_256_TIME	
(46C)	BITSTRING	1	TMA_DFHTASK_256_FLAG	
(46D)	UNSIGNED	3	TMA_DFHTASK_256_COUNT	
(470)	CHARACTER	8	TMA_DFHTASK_257	
(470)	UNSIGNED	4	TMA_DFHTASK_257_TIME	
(474)	BITSTRING	1	TMA_DFHTASK_257_FLAG	
(475)	UNSIGNED	3	TMA_DFHTASK_257_COUNT	
(478)	CHARACTER	8	TMA_DFHTASK_258	
(478)	UNSIGNED	4	TMA_DFHTASK_258_TIME	
(47C)	BITSTRING	1	TMA_DFHTASK_258_FLAG	

Offset Hex	Type	Len	Name (Dim)	Description
(47D)	UNSIGNED	3	TMA_DFHTASK_258_COUNT	
(480)	CHARACTER	8	TMA_DFHTASK_259	
(480)	UNSIGNED	4	TMA_DFHTASK_259_TIME	
(484)	BITSTRING	1	TMA_DFHTASK_259_FLAG	
(485)	UNSIGNED	3	TMA_DFHTASK_259_COUNT	
(488)	CHARACTER	8	TMA_DFHTASK_260	
(488)	UNSIGNED	4	TMA_DFHTASK_260_TIME	
(48C)	BITSTRING	1	TMA_DFHTASK_260_FLAG	
(48D)	UNSIGNED	3	TMA_DFHTASK_260_COUNT	
(490)	CHARACTER	8	TMA_DFHTASK_261	
(490)	UNSIGNED	4	TMA_DFHTASK_261_TIME	
(494)	BITSTRING	1	TMA_DFHTASK_261_FLAG	
(495)	UNSIGNED	3	TMA_DFHTASK_261_COUNT	
(498)	CHARACTER	8	TMA_DFHTASK_249	
(498)	UNSIGNED	4	TMA_DFHTASK_249_TIME	
(49C)	BITSTRING	1	TMA_DFHTASK_249_FLAG	
(49D)	UNSIGNED	3	TMA_DFHTASK_249_COUNT	
(4A0)	CHARACTER	8	TMA_DFHTASK_250	
(4A0)	UNSIGNED	4	TMA_DFHTASK_250_TIME	
(4A4)	BITSTRING	1	TMA_DFHTASK_250_FLAG	
(4A5)	UNSIGNED	3	TMA_DFHTASK_250_COUNT	
(4A8)	CHARACTER	8	TMA_DFHCICS_103	
(4A8)	UNSIGNED	4	TMA_DFHCICS_103_TIME	
(4AC)	BITSTRING	1	TMA_DFHCICS_103_FLAG	
(4AD)	UNSIGNED	3	TMA_DFHCICS_103_COUNT	
(4B0)	CHARACTER	8	TMA_DFHTERM_009	
(4B0)	UNSIGNED	4	TMA_DFHTERM_009_TIME	
(4B4)	BITSTRING	1	TMA_DFHTERM_009_FLAG	
(4B5)	UNSIGNED	3	TMA_DFHTERM_009_COUNT	
(4B8)	CHARACTER	8	TMA_DFHFILE_063	
(4B8)	UNSIGNED	4	TMA_DFHFILE_ 063_TIME	
(4BC)	BITSTRING	1	TMA_DFHFILE_063_FLAG	
(4BD)	UNSIGNED	3	TMA_DFHFILE_063_COUNT	
(4C0)	CHARACTER	8	TMA_DFHJOUR_010	
(4C0)	UNSIGNED	4	TMA_DFHJOUR_010_TIME	
(4C4)	BITSTRING	1	TMA_DFHJOUR_010_FLAG	
(4C5)	UNSIGNED	3	TMA_DFHJOUR_010_COUNT	
(4C8)	CHARACTER	8	TMA_DFHTEMP_011	
(4C8)	UNSIGNED	4	TMA_DFHTEMP_011_TIME	
(4CC)	BITSTRING	1	TMA_DFHTEMP_011_FLAG	
(4CD)	UNSIGNED	3	TMA_DFHTEMP_011_COUNT	
(4D0)	CHARACTER	8	TMA_DFHTERM_100	
(4D0)	UNSIGNED	4	TMA_DFHTERM_100_TIME	
(4D4)	BITSTRING	1	TMA_DFHTERM_100_FLAG	
(4D5)	UNSIGNED	3	TMA_DFHTERM_100_COUNT	
(4D8)	CHARACTER	8	TMA_DFHDEST_101	
(4D8)	UNSIGNED	4	TMA_DFHDEST_101_TIME	
(4DC)	BITSTRING	1	TMA_DFHDEST_101_FLAG	
(4DD)	UNSIGNED	3	TMA_DFHDEST_101_COUNT	
(4E0)	CHARACTER	8	TMA_DFHPROG_115	

Offset Hex	Type	Len	Name (Dim)	Description
(4E0)	UNSIGNED	4	TMA_DFHPROG_115_TIME	
(4E4)	BITSTRING	1	TMA_DFHPROG_115_FLAG	
(4E5)	UNSIGNED	3	TMA_DFHPROG_115_COUNT	
(4E8)	CHARACTER	8	TMA_DFHTASK_125	
(4E8)	UNSIGNED	4	TMA_DFHTASK_125_TIME	
(4EC)	BITSTRING	1	TMA_DFHTASK_125_FLAG	
(4ED)	UNSIGNED	3	TMA_DFHTASK_125_COUNT	
(4F0)	CHARACTER	8	TMA_DFHTASK_126	
(4F0)	UNSIGNED	4	TMA_DFHTASK_126_TIME	
(4F4)	BITSTRING	1	TMA_DFHTASK_126_FLAG	
(4F5)	UNSIGNED	3	TMA_DFHTASK_126_COUNT	
(4F8)	CHARACTER	8	TMA_DFHTASK_127	
(4F8)	UNSIGNED	4	TMA_DFHTASK_127_TIME	
(4FC)	BITSTRING	1	TMA_DFHTASK_127_FLAG	
(4FD)	UNSIGNED	3	TMA_DFHTASK_127_COUNT	
(500)	CHARACTER	8	TMA_DFHTASK_129	
(500)	UNSIGNED	4	TMA_DFHTASK_129_TIME	
(504)	BITSTRING	1	TMA_DFHTASK_129_FLAG	
(505)	UNSIGNED	3	TMA_DFHTASK_129_COUNT	
(508)	CHARACTER	8	TMA_DFHTASK_123	
(508)	UNSIGNED	4	TMA_DFHTASK_123_TIME	
(50C)	BITSTRING	1	TMA_DFHTASK_123_FLAG	
(50D)	UNSIGNED	3	TMA_DFHTASK_123_COUNT	
(510)	CHARACTER	8	TMA_DFHTERM_133	
(510)	UNSIGNED	4	TMA_DFHTERM_133_TIME	
(514)	BITSTRING	1	TMA_DFHTERM_133_FLAG	
(515)	UNSIGNED	3	TMA_DFHTERM_133_COUNT	
(518)	CHARACTER	8	TMA_DFHTERM_134	
(518)	UNSIGNED	4	TMA_DFHTERM_134_TIME	
(51C)	BITSTRING	1	TMA_DFHTERM_134_FLAG	
(51D)	UNSIGNED	3	TMA_DFHTERM_134_COUNT	
(520)	CHARACTER	8	TMA_DFHFEPI_156	
(520)	UNSIGNED	4	TMA_DFHFEPI_ 156_TIME	
(524)	BITSTRING	1	TMA_DFHFEPI_156_FLAG	
(525)	UNSIGNED	3	TMA_DFHFEPI_156_COUNT	
(528)	CHARACTER	8	TMA_DFHTASK_170	
(528)	UNSIGNED	4	TMA_DFHTASK_170_TIME	
(52C)	BITSTRING	1	TMA_DFHTASK_170_FLAG	
(52D)	UNSIGNED	3	TMA_DFHTASK_170_COUNT	
(530)	CHARACTER	8	TMA_DFHTASK_171	
(530)	UNSIGNED	4	TMA_DFHTASK_171_TIME	
(534)	BITSTRING	1	TMA_DFHTASK_171_FLAG	
(535)	UNSIGNED	3	TMA_DFHTASK_171_COUNT	
(538)	CHARACTER	8	TMA_DFHSYNC_173	
(538)	UNSIGNED	4	TMA_DFHSYNC_173_TIME	
(53C)	BITSTRING	1	TMA_DFHSYNC_173_FLAG	
(53D)	UNSIGNED	3	TMA_DFHSYNC_173_COUNT	
(540)	CHARACTER	8	TMA_DFHFILE_174	
(540)	UNSIGNED	4	TMA_DFHFILE_ 174_TIME	
(544)	BITSTRING	1	TMA_DFHFILE_174_FLAG	

Offset Hex	Type	Len	Name (Dim)	Description
(545)	UNSIGNED	3	TMA_DFHFILE_174_COUNT	
(548)	CHARACTER	8	TMA_DFHFILE_175	
(548)	UNSIGNED	4	TMA_DFHFILE_ 175_TIME	
(54C)	BITSTRING	1	TMA_DFHFILE_175_FLAG	
(54D)	UNSIGNED	3	TMA_DFHFILE_175_COUNT	
(550)	CHARACTER	8	TMA_DFHTASK_128	
(550)	UNSIGNED	4	TMA_DFHTASK_128_TIME	
(554)	BITSTRING	1	TMA_DFHTASK_128_FLAG	
(555)	UNSIGNED	3	TMA_DFHTASK_128_COUNT	
(558)	CHARACTER	8	TMA_DFHTASK_181	
(558)	UNSIGNED	4	TMA_DFHTASK_181_TIME	
(55C)	BITSTRING	1	TMA_DFHTASK_181_FLAG	
(55D)	UNSIGNED	3	TMA_DFHTASK_181_COUNT	
(560)	CHARACTER	8	TMA_DFHTASK_182	
(560)	UNSIGNED	4	TMA_DFHTASK_182_TIME	
(564)	BITSTRING	1	TMA_DFHTASK_182_FLAG	
(565)	UNSIGNED	3	TMA_DFHTASK_182_COUNT	
(568)	CHARACTER	8	TMA_DFHTASK_183	
(568)	UNSIGNED	4	TMA_DFHTASK_183_TIME	
(56C)	BITSTRING	1	TMA_DFHTASK_183_FLAG	
(56D)	UNSIGNED	3	TMA_DFHTASK_183_COUNT	
(570)	CHARACTER	8	TMA_DFHTASK_184	
(570)	UNSIGNED	4	TMA_DFHTASK_184_TIME	
(574)	BITSTRING	1	TMA_DFHTASK_184_FLAG	
(575)	UNSIGNED	3	TMA_DFHTASK_184_COUNT	
(578)	CHARACTER	8	TMA_DFHTEMP_178	
(578)	UNSIGNED	4	TMA_DFHTEMP_178_TIME	
(57C)	BITSTRING	1	TMA_DFHTEMP_178_FLAG	
(57D)	UNSIGNED	3	TMA_DFHTEMP_178_COUNT	
(580)	CHARACTER	8	TMA_DFHFILE_176	
(580)	UNSIGNED	4	TMA_DFHFILE_ 176_TIME	
(584)	BITSTRING	1	TMA_DFHFILE_176_FLAG	
(585)	UNSIGNED	3	TMA_DFHFILE_176_COUNT	
(588)	CHARACTER	8	TMA_DFHSYNC_177	
(588)	UNSIGNED	4	TMA_DFHSYNC_177_TIME	
(58C)	BITSTRING	1	TMA_DFHSYNC_177_FLAG	
(58D)	UNSIGNED	3	TMA_DFHSYNC_177_COUNT	
(590)	CHARACTER	8	TMA_DFHTASK_191	
(590)	UNSIGNED	4	TMA_DFHTASK_191_TIME	
(594)	BITSTRING	1	TMA_DFHTASK_191_FLAG	
(595)	UNSIGNED	3	TMA_DFHTASK_191_COUNT	
(598)	CHARACTER	8	TMA_DFHTASK_195	
(598)	UNSIGNED	4	TMA_DFHTASK_195_TIME	
(59C)	BITSTRING	1	TMA_DFHTASK_195_FLAG	
(59D)	UNSIGNED	3	TMA_DFHTASK_195_COUNT	
(5A0)	CHARACTER	8	TMA_DFHSYNC_196	
(5A0)	UNSIGNED	4	TMA_DFHSYNC_196_TIME	
(5A4)	BITSTRING	1	TMA_DFHSYNC_196_FLAG	
(5A5)	UNSIGNED	3	TMA_DFHSYNC_196_COUNT	
(5A8)	CHARACTER	8	TMA_DFHSOCK_241	

Offset Hex	Type	Len	Name (Dim)	Description
(5A8)	UNSIGNED	4	TMA_DFHSOCK_241_TIME	
(5AC)	BITSTRING	1	TMA_DFHSOCK_241_FLAG	
(5AD)	UNSIGNED	3	TMA_DFHSOCK_241_COUNT	
(5B0)	CHARACTER	8	TMA_DFHDATA_186	
(5B0)	UNSIGNED	4	TMA_DFHDATA_186_TIME	
(5B4)	BITSTRING	1	TMA_DFHDATA_186_FLAG	
(5B5)	UNSIGNED	3	TMA_DFHDATA_186_COUNT	
(5B8)	CHARACTER	8	TMA_DFHDATA_187	
(5B8)	UNSIGNED	4	TMA_DFHDATA_187_TIME	
(5BC)	BITSTRING	1	TMA_DFHDATA_187_FLAG	
(5BD)	UNSIGNED	3	TMA_DFHDATA_187_COUNT	
(5C0)	CHARACTER	8	TMA_DFHDATA_188	
(5C0)	UNSIGNED	4	TMA_DFHDATA_188_TIME	
(5C4)	BITSTRING	1	TMA_DFHDATA_188_FLAG	
(5C5)	UNSIGNED	3	TMA_DFHDATA_188_COUNT	
(5C8)	CHARACTER	8	TMA_DFHDATA_189	
(5C8)	UNSIGNED	4	TMA_DFHDATA_189_TIME	
(5CC)	BITSTRING	1	TMA_DFHDATA_189_FLAG	
(5CD)	UNSIGNED	3	TMA_DFHDATA_189_COUNT	
(5D0)	CHARACTER	8	TMA_DFHTASK_253	
(5D0)	UNSIGNED	4	TMA_DFHTASK_253_TIME	
(5D4)	BITSTRING	1	TMA_DFHTASK_253_FLAG	
(5D5)	UNSIGNED	3	TMA_DFHTASK_253_COUNT	
(5D8)	CHARACTER	8	TMA_DFHTASK_254	
(5D8)	UNSIGNED	4	TMA_DFHTASK_254_TIME	
(5DC)	BITSTRING	1	TMA_DFHTASK_254_FLAG	
(5DD)	UNSIGNED	3	TMA_DFHTASK_254_COUNT	
(5E0)	CHARACTER		TMA_USER_AREA	

The MN Domain Global Storage Area -- M N A --

- The domain status indication
- The storage subpool tokens
- The domain state lock tokens
- The TMA chain anchor
- The Monitoring Control Table names
- The Monitoring Control Table entry point and load address
- The Exception Record address
- The Performance Buffer address
- The SMF Buffer address
- The Sysevent Record address
- The Connector Sequences
- The Dictionary
- The MVS Workload Manager Token and PB array
- The Monitoring Status flags
- The Monitoring Catalogue record
- The Monitoring MAFPB address
- The Monitoring Statistics

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	344	MNA	
	Standard fields for restructured control blocks			
(0)	UNSIGNED	2	MNA_LENGTH	
(2)	CHARACTER	1	MNA_ARROW	
(3)	CHARACTER	3	MNA_DFH	
(6)	CHARACTER	2	MNA_DOMAIN	
(8)	CHARACTER	8	MNA_BLOCK_ID	

Offset Hex	Type	Len	Name (Dim)	Description
Current Monitoring Domain Status-initializing, initialized quiescing, quiesced, terminating or terminated.				
(10)	BITSTRING	2	MNA_DOMAIN_ STATUS	
(12)	CHARACTER	2	*	
Monitoring Status Flags READ THIS Do not change the offset within the MNA of the following MNA_ STATUS_FLAGS field. The inline macro DFHMNTST has a manually coded version of the MNA for testing the status of Monitoring from outside of the MN Domain.				
(14)	CHARACTER	4	MNA_STATUS_FLAGS	
(14)	BITSTRING	1	*	
	1...		MNA_CC_ERROR_FOUND	
	.1..		MNA_CC_UPDATE_REQUIRED	
	..1.		MNA_PA_ERROR_FOUND	
	...1		MNA_DICTIONARY_REQUIRED	
	.... 1...		MNA_MCT_ INITIALISED	
	.... .1..		MNA_MCT_LOADED	
	.... ..1.		MNA_MCT_DELETE	
	1		MNA_WLM_STATUS	
(15)	BITSTRING	1	*	
	1...		MNA_USER_EXIT_STATUS	
	.11.		*	
	...1		MNA_MCT_FIELDS_EXCLUDED	
	 1111		*	
(16)	BITSTRING	1	*	
	1...		MNA_EXCEPTION_STATUS	
	.1..		MNA_PERFORMANCE_STATUS	
	..1.		MNA_SYSEVENT_STATUS	
	...1		MNA_MONITORING_STATUS	
	.... 1...		MNA_SYNCPOINT_STATUS	
	.... .1..		MNA_CONVERSE_STATUS	
	.... ..1.		MNA_TIME	
	1		*	
(17)	UNSIGNED	1	MNA_CPU_TIMING	
Storage subpool tokens				
(18)	CHARACTER	8	MNA_CONTROL_POOL	Control subpool token
(20)	CHARACTER	8	MNA_TMA_ CELL_POOL	TMA subpool token
Monitoring Domain state lock token.				
(28)	ADDRESS	4	MNA_STATE_LOCK	
The number of TMAs currently allocated.				
(2C)	FULLWORD	4	MNA_CURRENT_TMAS	Current No of TMAs
Length of the standard TMA and the length of any TMA User Area (as defined by the MCT) for this execution of CICS.				
(30)	FULLWORD	4	MNA_TMA_LENGTH	
(34)	FULLWORD	4	MNA_TMA_USER_AREA_LENGTH	
Monitoring Control Table Name and Suffix				
(38)	CHARACTER	8	MNA_MCT_NAME	
(38)	CHARACTER	6	*	Currently loaded MCT
(3E)	CHARACTER	2	MNA_MCT_SUFFIX	Current MCT suffix
Entry Point of current MCT				
(40)	ADDRESS	4	MNA_MCT_ADDRESS	
(44)	ADDRESS	4	MNA_MCT_LOAD_ADDRESS	Load address of current MCT
length of currently loaded MCT. This field is zero if default MCT is being used.				
(48)	FULLWORD	4	MNA_MCT_LENGTH	
Monitoring Control Table Name and Suffix used when loading the MCT from the DFHRPL library.				
(4C)	CHARACTER	8	MNA_LOAD_ MCT_NAME	
(4C)	CHARACTER	6	*	
(52)	CHARACTER	2	MNA_LOAD_ MCT_SUFFIX	

Offset Hex	Type	Len	Name (Dim)	Description
	Exception Record Address			
(54)	ADDRESS	4	MNA_EXCEPTION_RECORD	
	Performance Buffer (PB) Management			
(58)	FULLWORD	4	MNA_PB_SIZE	PB size
(5C)	ADDRESS	4	MNA_PERFORMANCE_BUFFER	PB address
(60)	FULLWORD	4	MNA_PB_ LENGTH_LEFT	Amount free space left
(64)	ADDRESS	4	MNA_PB_NEXT_FREE	Next available space
(68)	FULLWORD	4	MNA_PD_RECORDS	No. Prfmnce Data records
(6C)	FULLWORD	4	MNA_PD_LENGTH	Prfmnce Data Record len
(70)	ADDRESS	4	MNA_PERFORMANCE_RECORD	Performance Data Record *
	Details of Monitoring Class Record(MCR) being written to SMF			
(74)	ADDRESS	4	MNA_RECORD_ ADDRESS	MCR address
(78)	FULLWORD	4	MNA_DATA_LENGTH	MCR length
(7C)	UNSIGNED	2	MNA_DATA_CLASS	MCR class
(7E)	CHARACTER	2	*	
	Response Codes (RC)			
(80)	CHARACTER	2	*	
(82)	UNSIGNED	1	MNA_LAST_SMF_RC	Last RC from SMF write
(83)	UNSIGNED	1	MNA_LAST_ SYSEVENT_RC	Last RC from SYSEVENT
	SMF Buffer Address - buffer includes storage for SMF header and product section.			
(84)	ADDRESS	4	MNA_SMF_BUFFER	
	Address of SYSEVENT record for writes to the MVS SRM.			
(88)	ADDRESS	4	MNA_SYSEVENT_ RECORD	
	Dictionary details			
(8C)	FULLWORD	4	MNA_DICTIONARY_ENTRIES	No of entries
(90)	FULLWORD	4	MNA_DICTIONARY_LENGTH	Length of Dictionary
(94)	ADDRESS	4	MNA_DICTIONARY_ PTR	Dictionary address
(98)	FULLWORD	4	MNA_DICTIONARY_USER_ENTRIES	Dictionary user entries
	Number and address of connectors in the output performance class record.			
(9C)	ADDRESS	4	MNA_OUT_CONNECTORS_PTR	
(A0)	FULLWORD	4	MNA_OUT_ CONNECTORS	
	Length of an individual connector, and length of storage required to hold a complete list of connectors.			
(A4)	FULLWORD	4	MNA_CONNECTOR_LENGTH	
(A8)	FULLWORD	4	MNA_CONNECTORS_LENGTH	
	MVS Workload Manager			
(AC)	BITSTRING	4	MNA_WLM_CONNECT_TOKEN	
(B0)	ADDRESS	4	MNA_WLM_PB_ARRAY_PTR	
(B4)	UNSIGNED	4	MNA_WLM_PB_ARRAY_SIZE	
(B8)	UNSIGNED	4	MNA_WLM_FREE_PERFORMANCE_BLK	
(BC)	UNSIGNED	4	MNA_WLM_MAX_PERFORMANCE_BLKS	
(C0)	UNSIGNED	4	MNA_WLM_ CURRENT_PERFORMANCE_BLKS	
(C4)	UNSIGNED	4	MNA_WLM_ MAX_SYS_PERFORMANCE_BLKS	
(C8)	UNSIGNED	4	MNA_WLM_ CUR_SYS_PERFORMANCE_BLKS	
(CC)	UNSIGNED	4	MNA_WLM_ NOTIFIED_MXT_VALUE	
	Frequency time and token for Timer calls			
(D0)	CHARACTER	4	MNA_FREQUENCY	

Offset Hex	Type	Len	Name (Dim)	Description
(D4)	CHARACTER	8	MNA_FREQUENCY_ TOKEN	
Frequency in progress indicator to prevent simultaneous frequency period intervals occurring.				
(DC)	BITSTRING	4	MNA_FREQUENCY_ IN_PROGRESS	
Subsystem id for SYSEVENT records				
(E0)	CHARACTER	8	MNA_SUBSYSTEM_ID	
Monitoring Catalogue Record				
(E8)	CHARACTER	64	MNA_CR	
Monitoring Authorised Facilities Parameter Block				
(128)	ADDRESS	4	MNA_MAFPB_PTR	
Global Statistics : Exception Records.				
(12C)	FULLWORD	4	MNA_EXCEPTION_ RECORDS	Num recs written
(130)	FULLWORD	4	MNA_EXCEPTION_ RECORDS_SUPP	Num recs suppressed
Performance Records.				
(134)	FULLWORD	4	MNA_PERFORMANCE_ RECORDS	Num recs written
(138)	FULLWORD	4	MNA_PERFORMANCE_ RECORDS_SUPP	Num recs suppressed
SMF Records.				
(13C)	FULLWORD	4	MNA_SMF_RECORDS	Num recs written
(140)	FULLWORD	4	MNA_SMF_ERRORS	Num Bad responses from SMF
SYSEVENT records				
(144)	FULLWORD	4	MNA_SYSEVENT_ RECORDS	Num recs written to SRM
(148)	FULLWORD	4	MNA_SYSEVENT_ ERRORS	Num bad responses
(14C)	FULLWORD	4	MNA_SYSEVENT_ RETRIES	NUM retries
Time (STCK) that global statistics were last reset				
(150)	CHARACTER	8	MNA_LAST_ RESET_TIME	
(158)	CHARACTER		*	

The MN Domain Catalog Record -- C A T A L O G --
The Monitoring Domain Catalog Record contains:
The Monitoring Control Table suffix
The Exception Class status
The Performance Class status
The Sysevent Class status
The Monitoring Class status
The Syncpoint monitoring status
The Converse monitoring status
The Mon clocks in GMT or LOCAL indicator
The Frequency monitoring time
The Subsystem id for Sysevent class records

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	64	DFHMNCR	
Monitoring Catalog Record.				
(0)	CHARACTER	2	MNCR_MCT_SUFFIX	MCT Suffix
Bit indicators of class settings and Monitoring global status.				
(2)	CHARACTER	1	MNCR_FLAGS	
Exception class ON/OFF Indicator.				
	1...		MNCR_EXCEPTION_ STATUS	
Performance class ON/OFF Indicator.				
	.1..		MNCR_PERFORMANCE_ STATUS	
SYSEVENT class ON/OFF Indicator.				

Offset Hex	Type	Len	Name (Dim)	Description
	..1.		MNCR_SYSEVENT_ STATUS	
	Monitoring global status ON/OFF indicator.			
	...1		MNCR_MONITORING_ STATUS	
	Syncpoint monitoring YES/NO indicator.			
	.... 1...		MNCR_SYNCPOINT_ STATUS	
	Converse monitoring YES/NO indicator.			
	.... .1..		MNCR_CONVERSE_ STATUS	
	Time in GMT/LOCAL indicator			
	.... ..1.		MNCR_TIME	
	1		*	
	Frequency monitoring time (packed)			
(3)	CHARACTER	4	MNCR_FREQUENCY	
	Subsystem id for Sysevent class			
(7)	CHARACTER	8	MNCR_SUBSYSTEM_ ID	
(F)	CHARACTER	8	*	
(17)	CHARACTER	41	*	

Constants

Len	Type	Value	Name	Description
8	CHARACTER	TMA	TMA_ID_STRING	
	MNA associated constants Eye catcher constants			
8	CHARACTER	ANCHOR	MNA_ID_STRING	
2	CHARACTER	MN	EYECATCHER_DOMID	
3	CHARACTER	DFH	EYECATCHER_DFH	
1	CHARACTER	>	EYECATCHER_ARROW	
	Subsystem name for SMF records			
4	CHARACTER	CICS	MNA_SUBSYSTEM_NAME	
	Storage Subpool ID strings			
8	CHARACTER	MN_CNTRL	CONTROL_POOL_NAME	
8	CHARACTER	MN_TMAS	TMA_CELL_POOL_NAME	
	Monitoring Domain Statuses			
2	DECIMAL	1023	MONITORING_ INITIALISING	
2	DECIMAL	1024	MONITORING_ INITIALISED	
2	DECIMAL	2047	MONITORING_ QUIESCING	
2	DECIMAL	2048	MONITORING_ QUIESCED	
2	DECIMAL	4095	MONITORING_ TERMINATING	
2	DECIMAL	4096	MONITORING_ TERMINATED	
	Monitoring Domain lock data			
8	CHARACTER	MN_GBLOK	STATE_LOCK_NAME	
	Monitoring Control Table Name			
8	CHARACTER	DFHMCT	MNA_DFHMCT	
	Monitoring Domain Exit Point Name			
8	CHARACTER	XMNOUT	MNA_EXIT_POINT	
	Monitoring Record Classes			
2	DECIMAL	1	MNA_DICTIONARY_ CLASS	
2	DECIMAL	3	MNA_PERFORMANCE_ CLASS	
2	DECIMAL	4	MNA_EXCEPTION_ CLASS	
	Performance Record Types			
4	CHARACTER	C	MNA_RECORD_ TYPE_CONVERSE	
4	CHARACTER	D	MNA_RECORD_ TYPE_DELIVER	
4	CHARACTER	F	MNA_RECORD_ TYPE_FREQUENCY	*
4	CHARACTER	S	MNA_RECORD_ TYPE_SYNCPOINT	*
4	CHARACTER	T	MNA_RECORD_ TYPE_TERMINATE	*
	CPU Timing constants			

Len	Type	Value	Name	Description
1	DECIMAL	1	MNA_CPU_START_REQUIRED	
1	DECIMAL	2	MNA_CPU_STARTED	
1	DECIMAL	3	MNA_CPU_STOP_REQUIRED	
1	DECIMAL	4	MNA_CPU_STOPPED	
Oddball constants				
0	BIT	1	MNA_ON	
0	BIT	0	MNA_OFF	
0	BIT	1	MNA_YES	
0	BIT	0	MNA_NO	
0	BIT	1	MNA_EXCEPTION_ON	
0	BIT	0	MNA_EXCEPTION_OFF	
0	BIT	1	MNA_PERFORMANCE_ON	
0	BIT	0	MNA_PERFORMANCE_ OFF	
0	BIT	1	MNA_SYSEVENT_ON	
0	BIT	0	MNA_SYSEVENT_OFF	
0	BIT	1	MNA_MONITORING_ON	
0	BIT	0	MNA_MONITORING_OFF	
0	BIT	1	MNA_SYNCPOINT_YES	*
0	BIT	0	MNA_SYNCPOINT_NO	*
0	BIT	1	MNA_CONVERSE_YES	*
0	BIT	0	MNA_CONVERSE_NO	*
0	BIT	1	MNA_TIME_LOCAL	*
0	BIT	0	MNA_TIME_GMT	*
4	HEX	0000000F	MNA_FREQUENCY_OFF	
declare frequency in progress and not in progress constants				
4	HEX	00000001	MNA_FIP_YES	
4	HEX	00000000	MNA_FIP_NO	
0	BIT	1	MNA_WLM_ENABLED	
0	BIT	0	MNA_WLM_DISABLED	
DUMP CODES				
8	CHARACTER	MN0001	MN_DUMP_ABEND_PROGRAM_CHECK	
8	CHARACTER	MN0002	MN_DUMP_SEVERE_ERROR	
8	CHARACTER	MN0003	MN_DUMP_INSUFFICIENT_STORAGE	
8	CHARACTER	MN0004	MN_DUMP_POSSIBLE_LOOP	
8	CHARACTER	MN0005	MN_DUMP_STORE_CLOCK_ERROR	
Message Numbers.				
4	DECIMAL	1	MNME_ABEND_PROGRAM_CHECK	
4	DECIMAL	2	MNME_SEVERE_ERROR	
4	DECIMAL	3	MNME_INSUFFICIENT_STORAGE	
4	DECIMAL	4	MNME_POSSIBLE_LOOP	
4	DECIMAL	5	MNME_STORE_CLOCK_ERROR	
4	DECIMAL	101	MNME_SMF_ERROR	
4	DECIMAL	102	MNME_SYSEVENT_ ERROR	
4	DECIMAL	103	MNME_MCT_NOT_FOUND	
4	DECIMAL	104	MNME_MCT_NOT_FOUND_IN_LIBRARY	
4	DECIMAL	105	MNME_USING_DEFAULT_MCT	
4	DECIMAL	106	MNME_CATALOGUE_READ_ERROR	
4	DECIMAL	107	MNME_CATALOGUE_UPDATE_ERROR	
4	DECIMAL	108	MNME_USING_MCT	
4	DECIMAL	109	MNME_MONITORING_ACTIVE	
4	DECIMAL	110	MNME_MONITORING_INACTIVE	
4	DECIMAL	111	MNME_SYSEVENT_ RETRY	*

NQA Enqueue domain anchor block

-

NQ domain anchor block (NQA)

This control block contains the global storage for the NQ domain.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	112	NQA	
(0)	CHARACTER	16	NQA_PREFIX	
(0)	UNSIGNED	2	NQA_LENGTH	Control block length
(2)	CHARACTER	14	NQA_EYECATCHER	>DFHNQANCHOR
(10)	CHARACTER	12	NQA_CHAIN_ POINTERS	
(10)	ADDRESS	4	NQA_FIRST_POOL	Head of pool chain
(14)	ADDRESS	4	NQA_FIRST_ BROWSE	Head of browse chain
(18)	ADDRESS	4	NQA_NQRNAME_ LIST	Head of nqrname list
(1C)	CHARACTER	32	NQA_SUBPOOLS	
(1C)	CHARACTER	8	NQA_GENERAL_ SUBPOOL	General subpool token
(24)	CHARACTER	8	NQA_NQPL_ SUBPOOL	NQPL subpool token
(2C)	CHARACTER	8	NQA_NQEA_ SUBPOOL	NQEA subpool token
(34)	CHARACTER	8	NQA_NQRN_ SUBPOOL	NQRN subpool token
(3C)	CHARACTER	8	NQA_LOCKS	
(3C)	ADDRESS	4	NQA_DOMAIN_LOCK	Domain lock token
(40)	ADDRESS	4	NQA_NQRNAME_ LOCK	nqrname lock token
(48)	CHARACTER	16	NQA_STATISTICS	
(48)	ADDRESS	4	NQA_STATS_ BUFFER_PTR	Address of statistics buffer
(4C)	ADDRESS	4	NQA_STATS_ BUFFER_LEN	Length of statistics buffer
(50)	CHARACTER	8	NQA_LAST_ RESET_TIME	Time of last statistics reset
(58)	CHARACTER	20	NQA_MISCELLANEOUS	
(58)	UNSIGNED	1	NQA_STATE	Enqueue domain state
(59)	CHARACTER	1	NQA_FLAGS	Flags
	1...		NQA_XRSINDI_ ACTIVE	Xrsindi exit active
	.111 1111		*	Reserved
(5A)	CHARACTER	2	*	Reserved
(5C)	FULLWORD	4	NQA_NUM_ ENQUEUE_POOLS	Number of enqueue pools
(60)	ADDRESS	4	NQA_DEFAULT_ INTERPRETER	Addr of default interpreter routine
(64)	CHARACTER	4	NQA_NQRN_ DIRECTORY	NQRN directory token
(68)	ADDRESS	4	NQA_DISPATCHER_ POOL	Addr of dispatcher pool
(70)	CHARACTER		NQA_END	Round to dword

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	NQA_INITIALISING	
1	DECIMAL	2	NQA_INITIALISED	
1	DECIMAL	3	NQA_QUIESCING	
1	DECIMAL	4	NQA_QUIESCED	
1	DECIMAL	5	NQA_TERMINATING	
1	DECIMAL	6	NQA_TERMINATED	

NQB Enqueue domain browse element

-

NQ domain browse element (NQB)

This control block represents a single enqueue browse. One of these control blocks exists for each enqueue browse that is in progress.

NQBs are chained together in a singularly linked list. The head of the list is in the NQA (anchor block).

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	NQB	
(0)	CHARACTER	84	NQB_PREFIX	
(0)	UNSIGNED	2	NQB_LENGTH	Control block length
(2)	CHARACTER	14	NQB_EYECATCHER	>DFHNQBROWSE
(10)	ADDRESS	4	NQB_NEXT_BROWSE_ELEMENT	Next browse element
(14)	ADDRESS	4	NQB_RMWT_BROWSE_TOKEN	Browse token of underlying RMWT browse
(18)	BITSTRING	1	NQB_FLAGS	
	1...		NQB_STABLE_ENQUEUES	Stable enqueues specified
	.1..		NQB_ENQSCOPE	Enqscope specified
(19)	CHARACTER	1	*	Reserved
(1A)	UNSIGNED	2	NQB_NAME_LENGTH	Length of name filter
(1C)	CHARACTER	4	NQB_SCOPE_ FILTER	Enqscope filter
(20)	CHARACTER	8	NQB_UOWID_ FILTER	Local uowid if browse filtered or nulls if not
(28)	CHARACTER	8	NQB_CURRENT_ UOWID	Local uowid of current UOW in RMWT browse
(30)	ADDRESS	4	NQB_CURRENT_UOW_TOKEN	UOW token of current UOW in RMWT browse
(34)	ADDRESS	4	NQB_OWNER_EXTENSION	Address of owner history extension for current UOW
(38)	ADDRESS	4	NQB_WAITER_EXTENSION	Address of waiter history extension
(3C)	ADDRESS	4	NQB_CURRENT_ENQUEUE_OWNER	UOW token of current enqueue being returned
(40)	ADDRESS	4	NQB_STABLE_NQEA	Last enq returned by STABLE_ENQUEUES browse
(44)	CHARACTER	4	NQB_BROWSING_ TRANID	Transaction id of txn performing the browse
(48)	CHARACTER	4	NQB_BROWSING_TRANNUM	Transaction number of txn performing the browse
(4C)	CHARACTER	8	NQB_BROWSING_TXN_TOKEN	Transaction token of txn performing the browse
(58)	CHARACTER	*	NQB_NAME_FILTER	Name filter

NQEA Enqueue domain queue element area

-

Queue Element Area (NQEA)

A single NQEA is used to represent each resource that is currently enqueued upon. Tasks waiting to gain control of a resource are also represented by an NQEA. A flag indicates whether the NQEA represents the resource owner or a task that is waiting for that resource.

Another flag indicates the scope (region or sysplex) of the enqueue.

Both owning and waiting NQEAs are chained from the 'NQ' work token in the UOW associated with them. Owning NQEAs are chained from the hash table in the NQPL (Enqueue Pool) that the resource belongs to. Waiting NQEAs are chained from the owning NQEA in FIFO order.

NQEAs that aren't in use are placed on a free chain anchored from their associated NQPL.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	NQEA	
(0)	CHARACTER	4	NQEA_PREFIX	
(0)	CHARACTER	4	NQEA_EYECATCHER	NQEA
(4)	CHARACTER	4	*	Overlaid fields
(4)	ADDRESS	4	NQEA_UOW_NEXT	Pointer to next NQEA owned or being waited on by UOW
(4)	ADDRESS	4	NQEA_NEXT_FREE	Next NQEA if on free chain
(8)	ADDRESS	4	NQEA_HASH_PREV	Previous NQEA on hash collision chain
(C)	ADDRESS	4	NQEA_HASH_NEXT	Next NQEA on hash collision chain
(10)	CHARACTER	16	NQEA_CLEARED_ FIELDS	Fields to cleared
(10)	ADDRESS	4	NQEA_NEXT_ WAITER	Chain of NQEAs waiting for this resource. Head of chain is the current owner
(14)	BITSTRING	1	NQEA_CLEARED_ FLAGS1	Various flags
	1...		NQEA_WAITER	0=owner , 1=waiter
	.1..		NQEA_RETAINED	0=active enqueue , 1=retained enqueue
	..1.		NQEA_SHUNT_OVERRIDE	0=use default shunt action 1=use override
	...1		NQEA_RESUME_REQUIRED	0=resume issued/not needed 1=resume required
	.... 1...		NQEA_NAME2_SUPPLIED	0=enqueue_name1 parm only 1=enqueue_name2 aswell
	.... .1..		NQEA_LONG_NAME	0=name length <= 256 chars 1=name length > 256
	.... ..1.		NQEA_OWNER_SHUNTED	0=owning uow not shunted 1=owning uow shunted
	1		NQEA_RESUME_FOR_LOCKED	0=no locked resume issued 1=resume because locked
(15)	BITSTRING	1	NQEA_CLEARED_ FLAGS2	Various flags
	1...		NQEA_SYSPLEX_ SCOPE	0=Region scope 1=Sysplex scope
	.1..		NQEA_SYSENQ_WAITING	0=not waiting 1=waiting Sysplex ENQ
	..1.		NQEA_SYSENQ_GRANTED	0=not granted 1=MVS enq granted
	...1 1111		*	Reserved
(16)	CHARACTER	2	*	Reserved
(18)	ADDRESS	4	NQEA_NQRMODEL_POINTER	Waiting nqrmodel
(1C)	FULLWORD	4	NQEA_TRANSACTION_COUNT	Number of times held with transaction duration
(20)	FULLWORD	4	NQEA_UOW_COUNT	Number of times held with UOW duration
(24)	CHARACTER	8	*	UOW associated with this owning/waiting NQEA
(24)	ADDRESS	4	NQEA_OWNER	Normally owner is kernel task addr
(24)	CHARACTER	8	NQEA_SHUNTED_ OWNER	If owner shunted then owner is the local uowid
(2C)	FULLWORD	4	NQEA_HASH_VALUE	Hash value of enqueue name

Offset Hex	Type	Len	Name (Dim)	Description
(30)	CHARACTER	4	NQEA_SUSPEND_ TOKEN	Suspend token if requester needs to wait
(34)	UNSIGNED	1	NQEA_SHUNT_ ACTION_OVERRIDE	Current shunt action if default has been overriden
(35)	BITSTRING	1	NQEA_PERMANENT_ FLAGS	Flags that aren't cleared
	1...		NQEA_QUICKCELLABLE	Eligible to be quickcelled
	.1..		NQEA_MVS_ GETMAINED	Storage obtained from MVS
	..11 1111		*	Reserved
(36)	CHARACTER	2	*	Reserved
(38)	FULLWORD	4	NQEA_LOCKED_ FAILURES	Number of times locked returned for this enqueue. Only valid when enqueue is in retained state
(40)	CHARACTER	8	*	Overlaid fields
(40)	CHARACTER	8	NQEA_ACTIVE_ START_TIME	Time enqueue obtained
(40)	CHARACTER	8	NQEA_WAIT_ START_TIME	Time enqueue wait started if waiting
(40)	CHARACTER	8	NQEA_RETAINED_ START_TIME	Time enqueue went into retained state if retained
(48)	ADDRESS	4	NQEA_POOL_ POINTER	NQPL that NQEA belongs to
(4C)	FULLWORD	4	NQEA_NAME2_ LENGTH	Length of enqueue_name2 parameter if supplied
(50)	CHARACTER	4	NQEA_ENQSCOPE	MVS enqscope name
(54)	CHARACTER	4	NQEA_SYSENQ_ECB	ECB used for ENQ macro@L1A
(58)	CHARACTER	4	NQEA_HASHMARK	Word which precedes name
(58)	FULLWORD	4	NQEA_NAME_ LENGTH	Length of enqueue name
(5C)	CHARACTER	*	NQEA_NAME	Start of Enqueue name

Constants

Len	Type	Value	Name	Description
4	DECIMAL	92	NQEA_FIXED_LENGTH	

NQOX Enqueue domain browse owner extension

-

NQ domain browse owner extension (NQOX)

This variable length vector is used to maintain a history of the enqueues names returned so far in the browse.

The start of the vector is used to store some names permanently for the duration of the browse.

After the permanent records are names that are stored temporarily for the current UOW in the browse.

The NQOX is addressed from the NQB (browse element) of the browse it relates to.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	NQOX	
(0)	CHARACTER	16	NQOX_PREFIX	
(0)	FULLWORD	4	NQOX_LENGTH	Control block length
(4)	CHARACTER	12	NQOX_EYECATCHER	>DFHNQOWNERX
(10)	ADDRESS	4	NQOX_SPARE_ NAME_STG_PTR	Address of spare name block storage
(14)	FULLWORD	4	NQOX_SPARE_ NAME_STG_LEN	Length of spare name block storage
(18)	FULLWORD	4	NQOX_MAXIMUM_ SLOTS	Number of slots in this extension
(1C)	FULLWORD	4	NQOX_TEMP_ SLOTS_USED	Number of temporary slots currently in use
(20)	FULLWORD	4	NQOX_PERM_ SLOTS_USED	Number of permanent slots in use for enqueues whose owner changed mid browse
(24)	CHARACTER	4	*	Reserved
(28)	CHARACTER	16	NQOX_OWNER_SLOT (*)	
(28)	ADDRESS	4	NQOX_ENQUEUE_ OWNER	UOW token of enqueue owner
(2C)	ADDRESS	4	NQOX_ENQUEUE_ POOL	Addr of enqueue pool
(30)	FULLWORD	4	NQOX_ENQUEUE_ NAME_LEN	Length of enqueue name
(34)	ADDRESS	4	NQOX_ENQUEUE_ NAME_PTR	Addr of enqueue name copy

Constants

Len	Type	Value	Name	Description
4	DECIMAL	16	NQOX_DEFAULT_ MAX_SLOTS	

NQPL Enqueue domain enqueue pool

-

Enqueue Pool control block (NQPL)

This control block represents a single enqueue pool. One of these control blocks exists for each enqueue pool that is created.

NQPL_SYSPLEX_SCOPE has been added to record the scope of enqueues in this pool.

NQPLs are chained together in a singularly linked list. The chain is ordered alphabetically by pool name. The head of the list is in the NQA.

For performance reasons the NQPL is divided into three separate separate sections. Ensure that new fields are added to the correct section of the control block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	384	NQPL	
(0)	CHARACTER	64	NQPL_SECTION_1	Performance sensitive
(0)	CHARACTER	4	NQPL_PREFIX	
(0)	CHARACTER	4	NQPL_EYECATCHER	NQPL
(4)	CHARACTER	8	NQPL_POOL_NAME	Name of enqueue pool
(C)	ADDRESS	4	NQPL_DOMAIN_LOCK_COPY	NQ domain lock token
(10)	CHARACTER	8	NQPL_FREE_NQEA_CHAIN	NQEA free chain
(10)	FULLWORD	4	NQPL_FIRST_CDS_COUNT	Free NQEA CDS count
(14)	ADDRESS	4	NQPL_FIRST_FREE_NQEA	First free NQEA for this pool
(18)	FULLWORD	4	NQPL_QUICKCELL_NAME_LENGTH	Max length of name in quickcelled NQEAs
(1C)	FULLWORD	4	NQPL_HASH_MASK	Masks hash value down to table index
(20)	FULLWORD	4	NQPL_HASH_ CONSTANT	Hashing constant
(24)	CHARACTER	28	NQPL_STATISTICS_ 1	Mainline statistics
(24)	FULLWORD	4	NQPL_TOTAL_REQUESTS	Number of enqueue requests in this pool
(28)	FULLWORD	4	NQPL_TOTAL_ BUSY	Number of times 'busy' returned
(2C)	FULLWORD	4	NQPL_TOTAL_ WAITED	Number of requests that have completed after waiting
(30)	CHARACTER	8	NQPL_TOTAL_WAITED_TIME	Time spent waiting by completed requests that waited.
(38)	CHARACTER	8	*	Pad to 64 byte boundary
(40)	CHARACTER	256	NQPL_SECTION_2	Hash table section
(40)	ADDRESS	4	NQPL_HASH_TABLE (0 63)	
(140)	CHARACTER	64	NQPL_SECTION_3	Non performance sensitive data
(140)	ADDRESS	4	NQPL_NEXT_POOL	Next pool in the chain
(144)	CHARACTER	1	NQPL_MISCELLANEOUS	
(144)	UNSIGNED	1	NQPL_DEFAULT_SHUNT_ACTION	Default action on shunt for enqueues in this pool
(145)	UNSIGNED	1	NQPL_ERROR_ LEVEL	Severity of response for errors using pool
(146)	UNSIGNED	1	NQPL_FLAGS1	miscellaneous flags
	1...		NQPL_SYSPLEX_ SCOPE	1=SYSPLEX scope, 0=REGION scope
	.1..		NQPL_DISPATCHER_TASK	1=DISPATCHER task, 0=UOW task
	..11 1111		*	Reserved
(147)	CHARACTER	5	*	Reserved
(14C)	CHARACTER	4	*	Reserved
(150)	CHARACTER	8	NQPL_ENQUEUE_INTERPRETATION	
(150)	UNSIGNED	1	NQPL_EXEC_INTERPRETER	How enqueues are to be interpreted by INQUIRE UOWENQ command
(151)	UNSIGNED	1	NQPL_DEFAULT_ TYPE	

Offset Hex	Type	Len	Name (Dim)	Description
				TYPE to be returned on INQUIRE UOWENQ by default interpreter only
(152)	CHARACTER	2	*	Reserved
(154)	ADDRESS	4	NQPL_INTERPRETER_ ADDR	Addr of interpreter routine for this pool
(158)	CHARACTER	40	NQPL_STATISTICS_ 2	Non mainline statistics
(158)	FULLWORD	4	NQPL_TOTAL_ LOCKED_IMMED	Number of times 'locked' returned immediately
(15C)	FULLWORD	4	NQPL_TOTAL_ LOCKED_WAITED	Number of times 'locked' returned after wait
(160)	FULLWORD	4	NQPL_TOTAL_ PURGED_CANCELLED	Number of times enqueue waiter cancelled
(164)	FULLWORD	4	NQPL_TOTAL_ PURGED_TIMED_OUT	Number of times enqueue waiter timed out
(168)	FULLWORD	4	NQPL_TOTAL_ RETAINED	Number of enqueues that HAVE been held in retained state
(16C)	CHARACTER	8	NQPL_TOTAL_ RETAINED_TIME	Time that enqueues were held in retained state
(174)	FULLWORD	4	NQPL_GLOBAL_ WAITED	Number of requests that have completed after wait for sysplex ENQ.
(178)	CHARACTER	8	NQPL_GLOBAL_ WAITED_TIME	Time spent waiting by completed requests that waited for sysplex ENQ.
(180)	CHARACTER		NQPL_END	Round to dword

Constants

Len	Type	Value	Name	Description
4	DECIMAL	64	NQPL_HASHSIZE	
4	DECIMAL	63	NQPL_HASHSIZE_ MINUS_1	
4	DECIMAL	63	NQPL_HASH_ MASK_VALUE	
Hash constant value				
4	DECIMAL	1904362337	NQPL_HASH_ CONSTANT_VALUE	
Enumerated values for nqpl_ default_ shunt_action				
1	DECIMAL	1	RELEASE_ENQUEUE	
1	DECIMAL	2	RETAIN_ENQUEUE	
1	DECIMAL	3	IGNORE_SHUNT	
Enumerated values for nqpl_ error_level				
1	DECIMAL	1	NQPL_RETURN_ EXCEPTION	
1	DECIMAL	2	NQPL_RETURN_ INVALID	
Enumerated values for nqpl_ exec_interpreter				
1	DECIMAL	1	NQPL_NO_INTERPRETATION	
1	DECIMAL	2	NQPL_DEFAULT_ INTERPRETATION	
1	DECIMAL	3	NQPL_OWN_ INTERPRETER	
Enumerated values for nqpl_ default_type				
1	DECIMAL	1	NQPL_TYPE_DATASET	
1	DECIMAL	2	NQPL_TYPE_EXECENQ	
1	DECIMAL	3	NQPL_TYPE_ EXECENQADDR	
1	DECIMAL	4	NQPL_TYPE_ EXECENQPLEX	
1	DECIMAL	5	NQPL_TYPE_FILE	
1	DECIMAL	6	NQPL_TYPE_TDQUEUE	
1	DECIMAL	7	NQPL_TYPE_TSQUEUE	
1	DECIMAL	8	NQPL_TYPE_ DISPATCHER	

NQWX Enqueue domain browse waiter extension

-

NQ domain browse waiter extension (NQWX)

This variable length vector is used to maintain a history of the UOW's that have so far been returned as waiters for the current enqueue in the browse.

The NQWX is addressed from the NQB (browse element) of the browse it relates to.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	NQWX	
(0)	CHARACTER	16	NQWX_PREFIX	
(0)	FULLWORD	4	NQWX_LENGTH	Control block length
(4)	CHARACTER	12	NQWX_EYECATCHER	>DFHNQWAITERX
(10)	FULLWORD	4	NQWX_MAXIMUM_ SLOTS	Number of slots in this extension
(14)	FULLWORD	4	NQWX_SLOTS_USED	Number of in-use slots
(18)	CHARACTER	8	NQWX_WAITER_SLOT (*)	
(18)	CHARACTER	8	NQWX_ENQUEUE_ WAITER	Local uowid of waiter

Constants

Len	Type	Value	Name	Description
4	DECIMAL	16	NQWX_DEFAULT_ MAX_SLOTS	

PAA Parameter manager domain anchor block

Segment Name= DFHPAA
DESCRIPTIVE NAME = CICS Parameter Manager (PA) Domain
Control Block declarations.
Function =
This file contains the control block and constant
declarations used by the Parameter Manager domain.
The file is included by each Parameter Manager domain
module.
The control blocks are:
DFHPAA - PA Anchor block.
PARM_SAVE_AREA - PA Override Save Area.
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
PA domain Anchor Block storage definition

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	80	DFHPAA	Anchor block
(0)	CHARACTER	16	PAA_PREFIX	Standard header
(0)	HALFWORD	2	PAA_LENGTH	Length of anchor block
(2)	CHARACTER	1	PAA_ARROW	Eyecatcher
(3)	CHARACTER	3	PAA_DFH	Eyecatcher
(6)	CHARACTER	2	PAA_DOMID	Domain Id
(8)	CHARACTER	8	PAA_BLOCK_NAME	Control block name
(10)	BITSTRING	1	PAA_DM_FLAGS	- Set by DFHPADM
	1...		CC_RECORD_ FOR_PA	Catalog record obtained?
	.1..		END_KEYWORD_ FOUND	Indicates if .END input
	..1.		PADM_ERROR_ RECOVERY	Error recovery entered
	...1		MORE_TO_ANALYSE	Unanalysed parms exist?
	.... 1...		INVALID_DATA	Inv. data found in DFHPASY
	.... .1..		SIT_LOADED	Indicates SIT been loaded
	.... ..1.		START_ALL	
	1		*	Spare
(11)	BITSTRING	1	PAA_IO_FLAGS	- Set by DFHPAIO
	1...		CONSOLE_FLAG	Input parms via Console?
	.1..		SYSIN_FLAG	Input parms via Sysin?
	..1.		SYSIN_EOF	Sysin end-of-file indicator
	...1		SYSIN_STATUS	Sysin open or closed?
	.... 1...		CONSOLE_ FIRST_RECORD	1st rec read from Console
	.... .1..		SYSIN_FIRST_ RECORD	1st record read from Sysin
	.... ..1.		OPENING_SYSIN	Footprints Sysin opening
	1		SYSIN_SAVED	Sysin saved in storage
(12)	BITSTRING	1	PAA_MORE_ IO_FLAGS	- Set by DFHPAIO
	1...		BRACKET_FOUND	Bracketted data flag
	.1..		QUOTE_FOUND	Quoted string flag
	..1.		MIXED_CASE	Mixed-case operand
(13)	UNSIGNED	1	START_SPECIFIED	Type of start
(14)	CHARACTER	8	SITNAME	Name of the loaded SIT
(14)	CHARACTER	6	*	Always DFHSIT
(1A)	CHARACTER	2	SIT_SUFFIX	Suffix of loaded SIT
(1C)	ADDRESS	4	PARM_SAVE_AREA_P	-> Override save area
(20)	ADDRESS	4	OVERRIDE_STORE_H	-> Temp stg for overrides
(24)	FULLWORD	4	OVERRIDE_STORE_L	Length of overrides so far
(28)	ADDRESS	4	ERRA_PTR	-> Kernel recovery area
(2C)	ADDRESS	4	SIT_PTR	-> SIT DSECT
(30)	ADDRESS	4	PASY_EP_PTR	-> DFHPASY entry point
(34)	CHARACTER	4	CATALOG_RECORD	PA catalog record
(38)	CHARACTER	8	APPLID	Applid for messages
(40)	CHARACTER	16	SYSIN_POINTERS	Chain of SYSIN records
(40)	ADDRESS	4	FIRST_POOL	-> First buffer pool
(44)	ADDRESS	4	CURRENT_POOL	-> Current buffer pool
(48)	ADDRESS	4	FIRST_REC	-> First record
(4C)	ADDRESS	4	CURRENT_REC	-> Current record
(50)	CHARACTER		*	End of PA anchor block

Parameter Manager Override Save Area

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	PARM_SAVE_AREA	PA Override Save Area
(0)	CHARACTER	16	PARM_SAVE_PREFIX	Standard header
(0)	HALFWORD	2	PARM_SAVE_ AREA_SIZE	Length of parm save area
(2)	CHARACTER	1	PARM_SAVE_ARROW	Eyecatcher
(3)	CHARACTER	3	PARM_SAVE_DFH	Eyecatcher
(6)	CHARACTER	2	PARM_SAVE_DOMID	Domain Id
(8)	CHARACTER	8	PARM_SAVE_ BLOCK_NAME	Control block name
(10)	HALFWORD	2	PARMS_LEN	Length of overrides
(12)	CHARACTER	*	PARMS	Overrides go here

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	DFHPAA_CR	The catalog record
(0)	CHARACTER	2	PA_CATALOG_ SUFFIX	SIT Suffix
(2)	CHARACTER	1	PA_RECORD_TYPE	STANDBY OR BLANK
(3)	CHARACTER	1	*	

Constants

Len	Type	Value	Name	Description
2	HEX	0101	TPID_PAGP_ENTRY	DFHPAGP Entry trace point
2	HEX	0102	TPID_PAGP_EXIT	DFHPAGP Exit trace point
2	HEX	0103	TPID_PAGP_BWTOR	DFHPAGP before WTOR
2	HEX	0104	TPID_PAGP_AWTOR	DFHPAGP after WTOR
2	HEX	0111	TPID_PAGP_INVDC	DFHPAGP inv domain call
2	HEX	0112	TPID_PAGP_ INV_FORMAT	DFHPAGP inv dom. format no.
2	HEX	0113	TPID_PAGP_ INV_FUNCTION	DFHPAGP inv function req.
2	HEX	0114	TPID_PAGP_INVRQDOM	DFHPAGP inv calling domain
2	HEX	0115	TPID_PAGP_INVSIT	DFHPAGP invalid SIT address
2	HEX	0116	TPID_PAGP_RECOVERY	DFHPAGP recovery entered
2	HEX	0201	TPID_PADM_ENTRY	DFHPADM Entry trace point
2	HEX	0202	TPID_PADM_EXIT	DFHPADM Exit trace point
2	HEX	0211	TPID_PADM_ INV_FORMAT	DFHPADM inv dom. format no.
2	HEX	0212	TPID_PADM_ INV_FUNCTION	DFHPADM inv function req.
2	HEX	0213	TPID_PADM_RECOVERY	DFHPADM recovery entered
2	HEX	0401	TPID_PASY_ENTRY	DFHPASY Entry trace point
2	HEX	0402	TPID_PASY_EXIT	DFHPASY Exit trace point
Messages - used when call is made to Message Domain.				
4	DECIMAL	1	MEID_RECOVERY	Msg DFHPA0001
4	DECIMAL	2	MEID_SEVERE_ERROR	Msg DFHPA0002
4	DECIMAL	4	MEID_LOOP	Msg DFHPA0004
4	DECIMAL	1924	MEID_LESSTHAN_ PARAMETER	Msg DFHPA1924
Dumpcodes - used when call is made to Message Domain.				
8	CHARACTER	PA0001	DUID_PA_RECOVERY	
8	CHARACTER	PA0002	DUID_PA_SEVERE_ ERROR	
8	CHARACTER	PA0004	DUID_PA_LOOP	
Constants				
1	CHARACTER	>	ARROW	Eyecatcher standard prefix
2	DECIMAL	120	BUFFER_SIZE	Size for Getmaining buffer
2	DECIMAL	4096	PAGE_SIZE	Size for Getmaining 1 page
2	DECIMAL	80	SYSIN_RECORD_L	Length of a SYSIN record.
4	DECIMAL	7	DWORDUP	Const to round up to dblwd
0	BIT	1	ON	Used for flag
0	BIT	0	OFF	manipulation.
0	BIT	1	YES	" "
0	BIT	0	NO	" "
0	BIT	1	OPEN	" "
0	BIT	0	CLOSED	" "
1	DECIMAL	0	WARM	Use Catalog
1	DECIMAL	1	COLD	¬Use catalog
4	HEX	FFFFFFF8	TURN_OFF_ LAST_3_BITS	
6	CHARACTER	DFHSIT	SIT_NAME	
7	CHARACTER	DFHPADM	PADM_NAME	
7	CHARACTER	DFHPAIO	PAIO_NAME	
7	CHARACTER	DFHPAGP	PAGP_NAME	
8	CHARACTER	DBDCCICS	DEFAULT_APPLID_ NAME	
1	CHARACTER	S	STANDBY	

PGA Macro save area

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	524	PESA	
(0)	CHARACTER	32	PESA_STANDARD	
(0)	CHARACTER	10	PESA_PREFIX	
(0)	HALFWORD	2	PESA_LENGTH	length for environment
(2)	CHARACTER	1	PESA_ARROW	>
(3)	CHARACTER	3	PESA_DFH	DFH
(6)	CHARACTER	4	PESA_BLOCK_ NAME	PESA
(A)	UNSIGNED	1	PESA_ENVIRONMENT_TYPE	the type of environment
(B)	CHARACTER	1	PESA_AMODE	save area for TCAAAM
(C)	ADDRESS	4	PESA_PREV	points to the previous
(10)	ADDRESS	4	PESA_EIS_APLI_SAVEAREA	for SYSTEM&PLT only being linked to
(14)	CHARACTER	12	PESA_PCTWA	Regs at time of link

Structure ends here for PESA_ENVIRONMENT_TYPEs of PESA_SYSTEM and PESA_PLT. Do not reference fields beyond this point for these types.

Offset Hex	Type	Len	Name (Dim)	Description
(20)	CHARACTER		PESA_STANDARD_ END	

This is the start of additional information which is stacked to allow EXEC CICS commands to be issued at the next link level.

Offset Hex	Type	Len	Name (Dim)	Description
(20)	CHARACTER	208	PESA_EXEC_ SPECIFIC	
(20)	ADDRESS	4	PESA_EISTG	Command level ASSEMBLER storage (TCAEISTG)
(24)	HALFWORD	2	PESA_CALEN	Commarea length EIBCALEN
(26)	CHARACTER	52	PESA_EIS_ EXEC_DATA	save area for the EIS
(5A)	CHARACTER	144	PESA_EIUS_ EXEC_DATA	save area for the EIUS
(EA)	CHARACTER	6	*	reserved

Structure ends here for PESA_ENVIRONMENT_TYPE of PESA_EXEC Do not reference fields beyond this point for this type.

Offset Hex	Type	Len	Name (Dim)	Description
(F0)	CHARACTER		PESA_EXEC_SPECIFIC_END	

This is the start of additional information which is stacked to allow EXEC CICS commands to be issued within EXEC CICS commands

Offset Hex	Type	Len	Name (Dim)	Description
(F0)	CHARACTER	240	PESA_SUPERLINK_SPECIFIC	
(F0)	CHARACTER	48	PESA_EIS_SUPERLINK_DATA	
(120)	CHARACTER	85	PESA_SYSTEM_EIB	
(175)	CHARACTER	16	PESA_EIUS_SUPERLINK_STACK	
(185)	CHARACTER	85	PESA_USER_EIB	
(1DA)	CHARACTER	1	PESA_TCAEISFL	reserved
(1DB)	CHARACTER	5	*	reserved

Structure ends here for PESA_ENVIRONMENT_TYPEs of PESA_TRUE and PESA_URM. Do not reference fields beyond this point for these types.

Offset Hex	Type	Len	Name (Dim)	Description
(1E0)	CHARACTER		PESA_SUPERLINK_SPECIFIC_END	

This is the start of additional information which is stacked to allow EXEC CICS commands to be issued within a limited subset of Global User Exits.

Offset Hex	Type	Len	Name (Dim)	Description
(1E0)	CHARACTER	44	PESA_GLUE_ SPECIFIC	
(1E0)	CHARACTER	36	PESA_COMMON_CONTROL_AREA	Communications TCACCCA
(204)	BITSTRING	1	PESA_EDF_REPLY	EDF reply byte (EISEDFRB)
(205)	CHARACTER	3	PESA_FLAGS	EIS flags
(205)	BITSTRING	1	PESA_FLAG2	(EISFLAG2)
(206)	BITSTRING	1	PESA_FLAG3	(EISFLAG3)
(207)	BITSTRING	1	PESA_FLAG5	(EISFLAG5)
(208)	CHARACTER	4	*	reserved

Structure ends here for PESA_ENVIRONMENT_TYPE of PESA_GLUE

Offset Hex	Type	Len	Name (Dim)	Description
(20C)	CHARACTER		PESA_END	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	12	LENGTH_TCAPCTWA	
4	DECIMAL	52	LENGTH_EISTACKA	
4	DECIMAL	48	LENGTH_EISUPERB	
4	DECIMAL	85	LENGTH_DFHEIBLK	
4	DECIMAL	144	LENGTH_EIUS_ STACK_AREA	
4	DECIMAL	16	LENGTH_EIUS_ SUPER_STACK	
Constants for pesa_ environment_type				
4	DECIMAL	1	PESA_EXEC	command level application
4	DECIMAL	2	PESA_GLUE	global user exit
4	DECIMAL	3	PESA_PLT	program list table program
4	DECIMAL	4	PESA_SYSTEM	CICS system program
4	DECIMAL	5	PESA_TRUE	task-related user exit
4	DECIMAL	6	PESA_URM	user-replaceable program
4	DECIMAL	240	PESA_LENGTH_EXEC	
4	DECIMAL	524	PESA_LENGTH_GLUE	
4	DECIMAL	32	PESA_LENGTH_PLT	
4	DECIMAL	32	PESA_LENGTH_SYSTEM	
4	DECIMAL	480	PESA_LENGTH_TRUE	
4	DECIMAL	524	PESA_LENGTH_URM	

PGDCC Program manager control blocks

Program Manager Anchor Block.
This control block contains the global storage for the
Program Manager domain.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	200	PGANCHOR	
(0)	CHARACTER	16	PGA_PREFIX	prefix
(0)	HALFWORD	2	PGA_LENGTH	inclusive length of anchor
(2)	CHARACTER	1	PGA_ARROW	>
(3)	CHARACTER	3	PGA_DFH	DFH
(6)	CHARACTER	2	PGA_DOMID	PG
(8)	CHARACTER	8	PGA_BLOCK_NAME	Anchor
(10)	CHARACTER	8	PGA_GENERAL_ SUBPOOL_TOKEN	PG general subpool token
(18)	CHARACTER	8	PGA_PPTE_ SUBPOOL_TOKEN	Program Definition subpool token
(20)	CHARACTER	8	PGA_JVMCLASS_ SUBPOOL_TOKEN	JVM class subpool token
(28)	CHARACTER	8	PGA_LLE_ SUBPOOL_TOKEN	Load List Element subpool token
(30)	CHARACTER	8	PGA_PGWE_ SUBPOOL_TOKEN	PG Wait Element subpool token
(38)	CHARACTER	8	PGA_HTB_ SUBPOOL_TOKEN	Handle Table Block subpool token
(40)	CHARACTER	8	PGA_HMRSA_ SUBPOOL_TOKEN	Handle Manager Register Save Area subpool token
(48)	CHARACTER	8	PGA_PTA_ SUBPOOL_TOKEN	Program Transaction area subpool token
(50)	CHARACTER	8	PGA_LAST_ RESET_TIME	time PG statistics last reset
(58)	ADDRESS	4	PGA_LOCK_TOKEN	PG domain lock token
(5C)	FULLWORD	4	PGA_PG_STATE	PG domain state
(60)	FULLWORD	4	PGA_AUTOINSTALL_ STATE	autoinstall state
(64)	FULLWORD	4	PGA_AUTOINSTALL_ CATALOG_STATE	autoinstall catalog state
(68)	CHARACTER	8	PGA_AUTOINSTALL_ EXIT_NAME	name of autoinstall user replaceable module

Offset Hex	Type	Len	Name (Dim)	Description
(70)	FULLWORD	4	PGA_ATTEMPTED_AUTOINSTALLS	number of attempted program autoinstalls
(74)	FULLWORD	4	PGA_REJECTED_AUTOINSTALLS	number of rejected program autoinstalls
(78)	FULLWORD	4	PGA_FAILED_AUTOINSTALLS	number of failed program autoinstalls
(7C)	ADDRESS	4	PGA_PPT_ DIRECTORY	Directory token for PPT
(80)	FULLWORD	4	PGA_PPT_VERSION_NUMBER	incremented each time PPT entry is discarded
(84)	CHARACTER	8	PGA_SYS_LLE_HEAD	head of system LLE chain
(8C)	CHARACTER	8	PGA_PGWE_HEAD	head of list of PGWEs
(94)	ADDRESS	4	PGA_SM_ ACCESS_TOKEN	access token for SMSRI INQUIRE_ACCESS
(98)	ADDRESS	4	PGA_SM_ISOLATION_TOKEN	isolation token for SMSRI SWITCH_SUBSPACE
(9C)	BITSTRING	1	PGA_INDICATORS	various flag bits
	1...		PGA_COLD_START	START=COLD in SIT
	.1..		PGA_STORAGE_PROTECT	result of SMSR INQUIRE_STORAGE_PROTECT
	..1.		PGA_PPT_RECOVERY_COMPLETE	PPT recovered from global catalog
	...1		PGA_XRSINDI_ ACTIVE	status of XRSINDI GLUE
	.... 1...		PGA_PG_ AVAILABLE	exec calls to PG valid
	.... .1..		PGA_LANGUAGES_AVAILABLE	languages establishment has been done so that autoinstall exit can be used
	11		*	reserved
(9D)	CHARACTER	3	*	reserved
(A0)	CHARACTER	4	PGA_LOCAL_SYSTEM_NAME	SYSIDNT value in SIT
(A4)	FULLWORD	4	* (9)	reserved
(C8)	CHARACTER		*	round to doubleword

Control Block Structure For Each Program Processing Table Entry.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	76	PPTE	
(0)	CHARACTER	44	PPTE_CATALOG_ RECORD	record written to the global catalog
(0)	CHARACTER	10	PPTE_PREFIX	eyecatcher
(0)	CHARACTER	1	PPTE_ARROW	
(1)	CHARACTER	3	PPTE_DFH	
(4)	CHARACTER	2	PPTE_DOMID	
(6)	CHARACTER	4	PPTE_BLOCK_ NAME	
(A)	HALFWORD	2	PPTE_LENGTH	
(C)	CHARACTER	8	PPTE_PROGRAM_ NAME	program name
(14)	UNSIGNED	1	PPTE_MODULE_ TYPE	module type: program mapset partitionset
(15)	UNSIGNED	1	PPTE_LANG_ DEFINED	program language passed to PGDD DEFINE_PROGRAM
(16)	UNSIGNED	1	PPTE_INSTALL_ TYPE	install type
(17)	BITSTRING	1	PPTE_DEFINITIONS	program definition bits
	1...		PPTE_CEDF_ STATUS	cedf status: ON cedf allowed OFF cedf inhibited
	.1..		PPTE_PROG_ ENABLED	avail status: ON enabled OFF disabled
	..1.		PPTE_ANY_ DATA_LOC	data location: ON any location OFF below 16M
	...1		PPTE_CICS_ EXEC_KEY	execution key: ON cics OFF user
	.... 1...		PPTE_DPLSUBSET	execution set: ON dplsubset OFF fullapi
	.... .1..		PPTE_RELOAD_ YES	reload status: ON load a new copy each use OFF do not reload
	.... ..1.		PPTE_REMOTE	remote definition: ON remote OFF local
	1		PPTE_DYNAMIC_STATUS	dynamic status ON dynamic DPL permitted OFF dynamic DPL not permitted
(18)	CHARACTER	1	PPTE_DEFINITIONS_ 2	more program definition bits
	1...		PPTE_THREADSAFE	concurrency of program as adjusted by APLI language establishment ON program is threadsafe OFF program is quasireentrant
	.1..		PPTE_DEFINED_THREADSAFE	concurrency of program as DEFINED ON program is threadsafe OFF program is quasireentrant

Offset Hex	Type	Len	Name (Dim)	Description
	..1.		PPTE_JVM	ON indicates program is to be run under JVM
	...1		PPTE_JVM_DEBUG	ON indicates JVM_DEBUG(YES) specified on definition
	 1111		*	reserved
(19)	CHARACTER	3	*	reserved
(1C)	CHARACTER	8	PPTE_REMOTE_ PROGID	remote program name
(24)	CHARACTER	4	PPTE_REMOTE_ SYSID	remote system name
(28)	CHARACTER	4	PPTE_REMOTE_ TRANID	server transaction name

Internals. This record is part of Program Manager's internal state data. It is never written to the global catalog and is always initialised when a new PPTE is created.

Offset Hex	Type	Len	Name (Dim)	Description
(2C)	CHARACTER	16	PPTE_INTERNALS	PG internal data
(2C)	ADDRESS	4	PPTE_LANG_TOKEN	language token
(30)	CHARACTER	4	PPTE_CS_WORD	word for Compare and Swap
(30)	UNSIGNED	1	PPTE_LANG_ DEDUCED	language as deduced by LE
(31)	UNSIGNED	1	PPTE_PROGRAM_ LOCK	program lock
(32)	BITSTRING	1	PPTE_INTERNAL_ FLAGS	
	1...		PPTE_ASSEMBLER_CICS	DFH assembler with no stub ON cics assembler program OFF normal program
	.111 1111		*	reserved
(33)	UNSIGNED	1	PPTE_RUNTIME_ENVIRONMENT	runtime environment JVM, LE/370, other
(34)	ADDRESS	4	PPTE_LOADER_ TOKEN	loader token
(38)	FULLWORD	4	PPTE_HOLD_COUNT	hold counter

Indicators. These are never written to the global catalog, and are always initialised when a new PPTE is created.

Offset Hex	Type	Len	Name (Dim)	Description
(3C)	CHARACTER	16	PPTE_INDICATORS	indicators
(3C)	FULLWORD	4	PPTE_USECOUNT	PG's usecount for programs that are not RELOAD(YES)
(40)	UNSIGNED	1	PPTE_LOAD_ STATUS	load status
(41)	BITSTRING	1	PPTE_INDICATOR_ FLAGS	
	1...		PPTE_CICS_HOLD	hold status: ON loaded for cics lifetime OFF loaded for task lifetime
	.1..		PPTE_PG_CATALOGED_PDB	did PG call LD to catalog Loader's program definition: ON yes PG did OFF no PG has not
	..1.		PPTE_PGWE	are there any wait elements for this program on the PGWE: ON >= 1 wait elements OFF 0 wait elements
	...1		PPTE_DELETE_IN_PROGRESS	has a delete_program started for this ppte. ON ==> locates finding this ppte must be suspended until the delete has completed, and then must be retried OFF ==> ppte is ok to use ppte_lock_owners_pta_ptr is set when this bit turned on
	.... 1...		PPTE_ADD_IN_PROGRESS	has an add_program started for this ppte. ON ==> locates finding this ppte must be suspended until the add has completed, and then must be retried OFF ==> ppte is ok to use ppte_lock_owners_pta_ptr is set when this bit turned on
	.... .111		*	reserved
(42)	CHARACTER	2	*	reserved
(44)	ADDRESS	4	PPTE_LOCK_OWNERS_PTA_PTR	pta_ptr of owner of a program lock. For diagnostic purposes only. Set when ppte_program_lock, ppte_add_in_progress or ppte_delete_in_progress is set. May be 0 if no pta associated with the request
(48)	ADDRESS	4	PPTE_JVM_ CLASS_PTR	address of JVM class data
(4C)	CHARACTER		*	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	258	PPTE_JVM_CLASS	
(0)	UNSIGNED	2	PPTE_JVM_CLASS_LENGTH	
(2)	CHARACTER	256	PPTE_JVM_ CLASS_DATA	

PTA - PG Transaction Area
This block contains the PG domain storage for a transaction
ALLOCATED : in DFHPGXM as part of PG INITIALIZE_ TRANSACTION
FREED : in DFHPGXM as part of PG TERMINATE_ TRANSACTION
WHERE : PGPTA subpool. Fixed length, CICS lifetime, CICS key, quickcell subpool.
HOW TO FIND : online it is addressed by the token returned by XM INQUIRE_ TRANSACTION_ TOKEN.

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(0)	STRUCTURE	66	PTA	
(0)	CHARACTER	16	PTA_PREFIX	
(0)	HALFWORD	2	PTA_LENGTH	
(2)	CHARACTER	1	PTA_ARROW	
(3)	CHARACTER	3	PTA_DFH	
(6)	CHARACTER	2	PTA_DOMID	
(8)	CHARACTER	8	PTA_BLOCK_NAME	
(10)	CHARACTER	8	PTA_TASK_ LLE_HEAD	
(18)	ADDRESS	4	PTA_PLCB_HEAD	-> highest logical level
(1C)	CHARACTER	28	PTA_XCTL_INFO	info from prepare xctl
(1C)	CHARACTER	8	PTA_XCTL_ PROGRAM_NAME	Name of prog for next XCTL
(24)	ADDRESS	4	PTA_XCTL_ PROG_PPTE	-> PPT entry for xctl
(28)	ADDRESS	4	PTA_XCTL_ LOAD_POINT	load point for xctl
(2C)	ADDRESS	4	PTA_XCTL_ ENTRY_POINT	entry point for xctl
(30)	FULLWORD	4	PTA_XCTL_ PROGRAM_LENGTH	program length for xctl
(34)	ADDRESS	4	PTA_XCTL_ LANGUAGE_TOKEN	language token for xctl
(38)	CHARACTER	8	PTA_LEVEL_COUNTS	level counters
(38)	FULLWORD	4	PTA_LOGICAL_ LEVEL	counts all levels
(3C)	FULLWORD	4	PTA_SYSTEMEXIT_ LEVEL	counts GLUEs and URMs
(40)	BITSTRING	1	PTA_FLAGS	flags
	1...		PTA_INPUTMSG_ RETURNED	inputmsg passed on RETURN
	.1..		PTA_PSEUDO_ CONV_COMMAREA	a pseudo-conversational commarea was passed to the first program in this transaction
	..1.		PTA_COMMAREA_ RETURNED	this transaction passed a valid commarea on a RETURN
	...1		PTA_AUTOINSTALL_ CALLED	running autoinstall exit, used to prevent recursion
	.... 1...		PTA_JVM_CALLED	a JVM program is running used to ensure no more than one JVM program active
	.... .111		*	reserved
(41)	UNSIGNED	1	PTA_HANDLE_ ABEND_CT	count of active handle abends
(42)	CHARACTER		*	

PLCB - PG Program Level Control Block.
This block contains the PG domain storage for a logical level
within a transaction.
ALLOCATED : as part of link to a logical level. There is no
explicit GETMAIN in PG because it resides in automatic storage.
FREED : on return from the logical level. There is no explicit
FREEMAIN in PG because it resides in automatic storage.
WHERE : automatic storage supplied by the Kernel.
HOW TO FIND : chained from the PTA for the transaction.
PLCBs are in a singly linked list.

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(0)	STRUCTURE	87	PLCB	
(0)	CHARACTER	16	PLCB_PREFIX	
(0)	HALFWORD	2	PLCB_LENGTH	
(2)	CHARACTER	1	PLCB_ARROW	
(3)	CHARACTER	3	PLCB_DFH	
(6)	CHARACTER	2	PLCB_DOMID	
(8)	CHARACTER	8	PLCB_BLOCK_NAME	
(10)	ADDRESS	4	PLCB_PREV	previous plcb
(14)	CHARACTER	28	PLCB_PROGRAM_ INSTANCE	instance of current prog
(14)	CHARACTER	8	PLCB_PROGRAM_ NAME	program name at this level
(1C)	ADDRESS	4	PLCB_PROG_PPTE	PPT entry for this level
(20)	CHARACTER	16	PLCB_PROGRAM_ DETAILS	This structure is used for improving performance
(20)	ADDRESS	4	PLCB_LOAD_ POINT	program load point
(24)	ADDRESS	4	PLCB_ENTRY_ POINT	program entry point
(24)	CHARACTER	1	*	
	1...		PLCB_AMODE_ 31	AMODE on=31 off=24

Offset Hex	Type	Len	Name (Dim)	Description
(28)	FULLWORD	4	PLCB_PROGRAM_ LENGTH	program length
(2C)	ADDRESS	4	PLCB_LANGUAGE_ TOKEN	program language extension
(30)	BITSTRING	1	PLCB_INSTANCE_ FLAGS	Bit settings are the same as those in PPTE_DEFINITIONS
	1...		PLCB_CEDF_ STATUS	CEDF status
	.1..		*	
	..1.		PLCB_ANY_ DATA_LOC	data location
	...1		*	
	.... 1...		PLCB_DPLSUBSET	program execution set
	.... .11.		*	
	1		PLCB_DYNAMIC_ STATUS	dynamic DPL status
(31)	CHARACTER	1	PLCB_ENVIRONMENT	environment information
(31)	UNSIGNED	1	PLCB_ENVIRONMENT_ TYPE	environment type
(32)	CHARACTER	2	*	reserved
(34)	ADDRESS	4	PLCB_HANDLE_ LEVEL_TKN	token identifying handle table at this level
(38)	CHARACTER	20	PLCB_COMMAREA_ INFO	commarea information
(38)	ADDRESS	4	PLCB_CA_CURRENT	current commarea address
(3C)	FULLWORD	4	PLCB_CA_ CURRENT_LEN	current commarea length
(40)	ADDRESS	4	PLCB_CA_LINK	commarea address on LINK to this level
(44)	FULLWORD	4	PLCB_CA_ LINK_LEN	commarea length on LINK to this level
(48)	BITSTRING	1	PLCB_CA_FLAGS	commarea flags
	1...		PLCB_CA_ CURRENT_X	current commarea exists
	.1..		PLCB_CA_COPY	current commarea is a copy
	..1.		PLCB_CA_ LINK_COPY	current commarea is a copy of the commarea passed on the LINK to this level
	...1		PLCB_CA_ READONLY	commarea passed on the LINK is in readonly storage
	 1111		*	reserved
(49)	CHARACTER	1	PLCB_CA_ STORAGE_CLASS	CICS,CICS24,USER,USER24 only valid when plcb_ca_copy is set
(4A)	CHARACTER	2	*	reserved
(4C)	CHARACTER	8	PLCB_INVOKING_ PROG	invoking program name
(54)	HALFWORD	2	PLCB_EXIT_NUMBER	number which identifies a Global User Exit point
(56)	BITSTRING	1	PLCB_FLAGS	
	1...		PLCB_INPUTMSG_ SUPPLIED	inputmsg passed on LINK or XCTL to this level
	.1..		PLCB_XCTL_ IN_PROGRESS	XCTL in progress
	..1.		PLCB_HANDLE_ ABEND_PGM	abend handler program
	...1		PLCB_SYSEIB_ REQUEST	SYSEIB specified
	.... 1...		PLCB_HPJ_ PROGRAM	JAVA (HPJ) program
	.... .111		*	reserved
(57)	CHARACTER		*	

PGWE
The PGWE represents a task which is attempting to acquire the program lock. If the program lock is locked, the PGWE is added to the PGWE chain and the task is suspended.
ALLOCATED : when Program Manager attempts to obtain the program lock.
FREED : when the lock is obtained successfully.
WHERE : obtained from the pgwe subpool.
HOW TO FIND : elements are chained to the PGWE chain anchored in the PG anchor block by pga_ pgwe_head.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	PGWE	
(0)	CHARACTER	8	PGWE_PREFIX	
(0)	ADDRESS	4	PGWE_NEXT	
(4)	ADDRESS	4	PGWE_PREV	set to 0 when remove from queue
(8)	BITSTRING	4	PGWE_SUSPEND_ TOKEN	
(C)	ADDRESS	4	PGWE_PPTE_PTR	
(10)	CHARACTER	8	PGWE_PROGRAM_ NAME	
(18)	CHARACTER		*	

LLE
A Load List Element represents an instance of a program that
has been explicitly loaded.
ALLOCATED : when a program is explicitly loaded
FREED : when a program is explicitly released, or at end of
task for programs loaded for the lifetime of the task.
WHERE : obtained from the lle subpool
HOW TO FIND : elements are chained to the system LLE chain
anchored in the PG anchor block or the task LLE chain anchored
in the Program Transaction Area.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	LLE	
(0)	CHARACTER	8	LLE_PREFIX	
(0)	ADDRESS	4	LLE_NEXT	
(4)	ADDRESS	4	LLE_PREV	
(8)	ADDRESS	4	LLE_PPTE_ADDRESS	
(C)	ADDRESS	4	LLE_INSTANCE	
(10)	CHARACTER		*	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	1	PGA_INITIALISING	
4	DECIMAL	2	PGA_INITIALISED	
4	DECIMAL	3	PGA_QUIESCING	
4	DECIMAL	4	PGA_QUIESCED	
4	DECIMAL	5	PGA_TERMINATING	
4	DECIMAL	6	PGA_TERMINATED	
Values for pga_ autoinstall_ state.				
4	DECIMAL	0	PGA_DISABLED	
4	DECIMAL	1	PGA_ENABLED	
Values for pga_ autoinstall_ catalog_state.				
4	DECIMAL	1	PGA_CATALOG_ALL	
4	DECIMAL	2	PGA_CATALOG_MODIFY	
4	DECIMAL	3	PGA_CATALOG_NONE	
Miscellaneous Constants.				
10	CHARACTER	>DFHPGPPTE	PPTE_PREFIX_VALUE	
1	CHARACTER	>	PPTE_ARROW_VALUE	
3	CHARACTER	DFH	PPTE_DFH_VALUE	
2	CHARACTER	PG	PPTE_DOMID_VALUE	
4	CHARACTER	PPTE	PPTE_BLOCK_ NAME_VALUE	
Declarations For Program Lock.				
4	DECIMAL	1	PPTE_LOCKED	
4	DECIMAL	2	PPTE_UNLOCKED	
Declarations For Module Types.				
4	DECIMAL	1	PPTE_PROGRAM	
4	DECIMAL	2	PPTE_MAPSET	
4	DECIMAL	3	PPTE_PARTITIONSET	
Declarations For Type Of PPTE Installation.				
4	DECIMAL	1	PPTE_BUILT_ FROM_RDO	
4	DECIMAL	2	PPTE_BUILT_ FROM_CATALOG	
4	DECIMAL	3	PPTE_BUILT_ FROM_GROUPLIST	
4	DECIMAL	4	PPTE_AUTOINSTALL	
4	DECIMAL	5	PPTE_SYSTEM_ AUTOINSTALL	
4	DECIMAL	6	PPTE_MANUAL	
Declarations For Load Status.				
4	DECIMAL	1	PPTE_LOADABLE	
4	DECIMAL	2	PPTE_NOT_LOADABLE	
4	DECIMAL	3	PPTE_NOT_LOADED	

Len	Type	Value	Name	Description
Language Name Declarations. Values are declared here for both the language as defined by the caller of PGDD DEFINE_ PROGRAM and as deduced by LE. The ppte_ lang_defined cannot have the value ppte_ not_deduced or ppte_ cobol2 The ppte_ lang_defined value of ppte_ not_defined means that the program was EXEC LOADed, and language establishment could not find any language. The program is usually treated as not deduced. It is separated from not deduced so that language establishment is only done once. The following equates to apli values are done to improve performance. The ppte_ not_deduced value has no meaning to apli. The value of 255 is used as it is cannot be given by CDURUN.				
4	DECIMAL	1	PPTE_NOT_DEFINED	not def'd by user
4	DECIMAL	255	PPTE_NOT_DEDUCED	not deduced by LE
4	DECIMAL	2	PPTE_ASSEMBLER	(or ada)
4	DECIMAL	4	PPTE_C370	
4	DECIMAL	3	PPTE_COBOL	
4	DECIMAL	7	PPTE_COBOL2	
4	DECIMAL	5	PPTE_LE370	le370 (or C++)
4	DECIMAL	6	PPTE_PLI	PL/I
4	DECIMAL	9	PPTE_JVM_LANG	JVM
Runtime Environment Name Declarations				
4	DECIMAL	1	PPTE_JVM_RUNTIME	
4	DECIMAL	2	PPTE_LE370_RUNTIME	
4	DECIMAL	3	PPTE_NON_ LE370_RUNTIME	
Constants for plcb_ environment_type. The following equates to apli values are done to improve performance.				
4	DECIMAL	2	PLCB_EXEC	command level application
4	DECIMAL	5	PLCB_GLUE	global user exit
4	DECIMAL	6	PLCB_PLT	program list table program
4	DECIMAL	1	PLCB_SYSTEM	CICS system program
4	DECIMAL	4	PLCB_TRUE	task-related user exit
4	DECIMAL	3	PLCB_URM	user-replaceable program

PGHM Handle manager declarations

```
===================================================================
 Handle Table Block
 The Handle Manager owns and manages the repository of the data
 which needs to be held to record a user program's EXEC CICS Handle
 requests.
 Data for each unique Condition, AID or Abend is retained as a
 single entry in the repository: an entry in this repository is
 known as a Handle Table Entry. There are three such tables of
 entries: The Conditions Table which contains the entries for all
 handled Conditions, the AIDs Table which contains the entries for
 all handled AIDs and the Abend Table which contains the entry -
 there can only be one entry in this table - for a handled Abend.
 In addition, 16 bits are set aside in the Block to hold a set of
 flags used to indicate whether any of the following conditions
 have been handled by the user: RDATT, WRBRK, EOF, SIGNAL,
 OVERFLOW, NOSPACE, QBUSY, NOSTG, ENQBUSY, NOJBUFSP, SYSBUSY and
 SESSBUSY. These flags are used by various EXEC CICS API handling
 modules and are provided to improve run-time performance in their
 respective areas.
 A Handle Table Block therefore holds all data representing a
 single level of the handle state. A multi-level handling system
 is enabled with this technique because the current Handle Table
 Block can be stacked at any time, for example as a result of a
 PUSH command, and a new level instated: similarly, a previous
 level can be reinstated following a POP.
 Addressability to the current Handle Table Block is via a pointer
 named the Handle Level Token which is defined in the Program
 Level Control Block owned by the PG Domain. The Program Level
 Control Block is addressed via the PG Domain Transaction Storage
 which is in turn anchored off the PG Transaction Token,
 managed by the Transaction Manager. The Handle Manager obtains
 addressability to the PG Token and thus to the Handle Level Token
 using the DFHXMIQ Inquire_Transaction_Token service.
 Whenever a Handle Table Block is PUSHed onto the stack and a new
 Block created, the new Block contains a pointer, in its
 htb_prev_table field, to the PUSHed Block. This both facilitates
 the reinstatement of the previous Block if a POP is driven, but
 also allows for the speedy freeing up of all Handle Table Blocks
 at program termination.
 A Handle Table Block is acquired out of the HTB subpool.
===================================================================
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	1936	HTB	
(0)	CHARACTER	16	HTB_PREFIX	
(0)	HALFWORD	2	HTB_LENGTH	
(2)	CHARACTER	1	HTB_ARROW	
(3)	CHARACTER	3	HTB_DFH	
(6)	CHARACTER	2	HTB_DOMID	
(8)	CHARACTER	8	HTB_HTB	
(10)	CHARACTER	8	*	
(10)	ADDRESS	4	HTB_PREV_TABLE	address of previous table/zero
(14)	ADDRESS	4	HTB_USED_RSAS	address of 1st in use RSA
(18)	CHARACTER	1912	HTB_TABLES	
(18)	CHARACTER	4	*	
(18)	BITSTRING	2	FASTPATH_FLAGS	
(1A)	BITSTRING	2	*	Conditions table
(1C)	CHARACTER	1416	HTB_CONDITIONS_ TABLE	
				AIDs table
(5A4)	CHARACTER	480	HTB_AIDS_TABLE	Abend table
(784)	CHARACTER	12	HTB_ABEND_TABLE	

```
=====================================================================
  Handle Table Entry
  An unique entry exists in the appropriate table for every
  possible condition, AID or abend.
  Handle Condition entries are held within the table known as
htb_ conditions_ table: Handle AID entries are held within the
htb_  aids_table: and the single Handle Abend entry is held in
htb_  abend_table. All three tables form part of the current
  Handle Table Block.
The first byte of every entry - named HTE_  ACTIVE - is used to
  denote whether or not that particular entry is active, ie that
  some user handle for that condition, AID or abend has been issued
  at the current level.
Should HTE_  ACTIVE be 00, ie FALSE, then the entry is not active.
For any value of HTE_  ACTIVE other than 00, the entry IS active.
=====================================================================
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	HTE	
(0)	BITSTRING	1	HTE_ACTIVE	0 = entry is not active ¬0 = this entry is active
	1...		HTE_DEFAULT	... take system default
	.1..		HTE_IGNORE	... ignore the event
	..1.		HTE_ABEND_ PROGRAM	handle abend(program)
	...1 1111		*	the 'depending on' value
(1)	BITSTRING	1	HTE_LANGUAGE	the language of the program issuing the handle
(2)	BITSTRING	1	HTE_PROGRAM_MASK	the program mask of the program issuing the handle
(3)	BITSTRING	1	HTE_EXECUTION_ KEY	the execution key of the program issuing the handle
(4)	CHARACTER	8	HTE_PROGRAM	handle abend program name
(4)	CHARACTER	4	HTE_LABEL	handle go to label address
(4)	ADDRESS	4	HTE_COBOL_RSA	RSA address (Cobol only)
(4)	CHARACTER	1	HTE_LABEL_ BYTE	
	1...		HTE_LABEL_ AMODE_31	AMODE on=31 off=24
(8)	ADDRESS	4	HTE_USER_RSA	caller's RSA address

```
 -

  Program Manager Transaction Token

  This is a special token, managed by the Transaction Manager, and
  owned by the PG Domain.

  The Handle Manager will use this token in order to find the
  address of the PG Domain's transaction storage: this latter area
  contains the Handle Level Token which is used by the Handle
  Manager to access the current Handle Table Block.

  The PG Transaction Token is accessed by the Handle Manager using
  the DFHXMIQ Inquire_Transaction_Token service.
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	PG_TRANSACTION_ TOKEN	
(0)	FULLWORD	4	*	
(4)	ADDRESS	4	TRANSACTION_ STG_PTR	

```
==================================================================
  Handle Manager Register Save Area
  A register save area has to be acquired by CICS during the
  processing of Handle requests for Cobol programs: the area is
  needed to hold the contents of the user's registers as at the
  time of the Handle command. These register values remain
  unchanged for the duration of that handle, and do not alter for
  any intervening EXEC CICS commands.
  The Handle registers are necessary because, in the case of Cobol
  programs only, when a handled event occurs, CICS passes control
  back to the program instruction immediately following the Handle:
  this instruction is a Cobol 'goto lab1, lab2.... depending on
  dfheigdi' statement and it needs the register values at the
  original handle in order to operate correctly.
  A single register save area is acquired when needed out of the
  HMRSA subpool. Every distinct event within a single command is
  able to share the same registers, therefore in order to assist
  with the management of the save areas, a count is maintained for
  each area. For every event in any one Handle command the
rsa_  user_count field is incremented by one. Whenever a new
  handle for an event is issued, thereby rendering the first save
  area unwanted for that event, the count is decremented. When the
  count reaches zero, the register save area is returned to the
subpool.
  Register save areas are chained together so that those in use may
  be speedily freed during program termination.
==================================================================
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	72	RSA	
(0)	CHARACTER	64	RSA_REGS	
(40)	FULLWORD	4	RSA_USER_COUNT	
(44)	ADDRESS	4	RSA_NEXT	

PRS Partner domain static storage area

```
CONTROL BLOCK NAME = DFHPRSPS
DESCRIPTIVE NAME = CICS Partner Static Storage Area
FUNCTION =
     This control block provides the global information for
     the Partner Resource Manager which must be around for
     the duration of the CICS execution.
     It contains:
       Partner Resource Manager subpool token
       Partner Resource Manager initialization suspend token
       Partner Resource Manager status
       Addresses of Partner Resource Manager gates
LIFETIME =
     The control block is created during CICS initialization
     by DFHPRIN1, and exists for as long as the CICS system.
STORAGE CLASS =
     The control block is in subpool DFHAPDAN. The token for
     this subpool is stored in the CSA optional features list
     in field CSADSANT.
LOCATION =
     The Partner Static Area is located by field SSZPRM in
     the static storage address list.
INNER CONTROL BLOCKS = None
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS = None
 MODULE TYPE = Control block definition
EXTERNAL REFERENCES = None
  DATA AREAS = None
  CONTROL BLOCKS = None
  GLOBAL VARIABLES (Macro pass) = None
     PARTNER STATIC STORAGE AREA
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	44	PRM_SSA	
Block prefix				
(0)	CHARACTER	16	PREFIX	block prefix area
(0)	HALFWORD	2	BLOCK_LENGTH	block length
(2)	CHARACTER	1	ARROW	'>'
(3)	CHARACTER	3	DFH	'DFH'
(6)	CHARACTER	2	DOMID	'PR'
(8)	CHARACTER	8	BLOCK_NAME	'PRSTATIC'
Block body				
(10)	CHARACTER	28	BODY	body of block
Partner Resource Manager fields				
(10)	CHARACTER	16	*	
(10)	ADDRESS	4	INIT_SUSPEND_ TOKEN	Suspend token
(14)	CHARACTER	8	SUBPOOL_TOKEN	Partner Resource Manager's subpool token
(1C)	UNSIGNED	1	INIT_STATUS	Status of Partner Resource Manager
(1D)	CHARACTER	3	*	Reserved
Partner Resource Manager entry points				
(20)	CHARACTER	12	*	
(20)	ADDRESS	4	PRPT_GATE	Gate PRPT
(24)	ADDRESS	4	PRFS_GATE	Gate PRFS
(28)	ADDRESS	4	PRCM_GATE	Gate PRCM

Constants

Len	Type	Value	Name	Description
1	DECIMAL	44	PRM_SSA_LENGTH	
Constants representing status of Partner Resource Manager initialization				
2	DECIMAL	1	PRM_STATIC_ STORAGE_INITIALIZED	
2	DECIMAL	2	PRM_ACQUIRE_ SUSPEND_TOK_FAILED	
2	DECIMAL	3	PRM_ACQUIRED_ SUSPEND_TOK	
2	DECIMAL	4	PRM_INIT_ TASK_ATTACHED	
2	DECIMAL	5	PRM_INIT_ TASK_STARTED	
2	DECIMAL	6	PRM_LOAD_ PRPT_FAILED	
2	DECIMAL	7	PRM_LOADED_PRPT	
2	DECIMAL	8	PRM_LOAD_ PRFS_FAILED	
2	DECIMAL	9	PRM_LOADED_PRFS	
2	DECIMAL	10	PRM_LOAD_ PRCM_FAILED	
2	DECIMAL	11	PRM_LOADED_PRCM	
2	DECIMAL	12	PRM_LOAD_ PRRP_FAILED	
2	DECIMAL	13	PRM_LOADED_PRRP	
2	DECIMAL	14	PRM_PARTNER_ RECOVERY_FAILED	
2	DECIMAL	15	PRM_PARTNER_ RECOVERED	
2	DECIMAL	16	PRM_INIT_SUCCEEDED	
2	DECIMAL	17	PRM_OPEN_ FOR_BUSINESS	
Block name for PR static				
8	CHARACTER	PRSTATIC	PRM_SSA_BLOCK_ NAMEI	

PTE Partner table entry

```
==================================================================
 CONTROL BLOCK NAME = DFHPTEPS
 DESCRIPTIVE NAME = CICS (PARTNER)
                Partner Table Entry
 FUNCTION =
     Defines the layouts of entries in the Partner Table,
     as it exists both in main storage and in the CICS catalog.
     The Partner Table is owned by the Partner component, also
     called the Partner Resource Manager, which encapsulates
     all accesses to the table.
     The Partner Table is the CICS implementation of the Side
     Information Table introduced by SAA CPI-C. (See the SAA
     CPI Communications Reference for details.) Each entry in
     the Partner Table contains information needed to
     initialize a conversation with a partner program on a
     remote LU, which can thus be specified by the application
     be specifying only the name of the entry (known as the
     sym_dest_name).
     An entry in the Partner Table contains the following
     pieces of information:
     - partner_LU_name
       indicates the name of the LU where the partner program
       is located. It can be either a simple network LU
       name, or netname, of one to eight characters, or else
       a fully qualified name of the form network.netname
       where network is a one to eight character network id
       and netname is a one to eight character network LU
       name.
     - profile_name
       the name of the CICS communication profile. This
       profile contains a mode_name which is used to designate
       the properties for the session which will be allocated
       for the conversation.
     - TP_name
       the name of the remote transaction program.
     Note that this implementation accesses the mode_name of
     the side information indirectly via the CICS profile.
 LIFETIME =
     PTEs are created and destroyed only via the PRPT gate of
     the Partner Resource Manager, module DFHPRPT.
 STORAGE CLASS =
     Storage for PTEs is drawn from a subpool created by
     DFHPRRP for this sole purpose.
 LOCATION =
     PTEs are located via scatter tables managed by DFHTMP.
 INNER CONTROL BLOCKS =
     None.
 NOTES :
  DEPENDENCIES = S/370
  RESTRICTIONS =
       None.
  MODULE TYPE = Control block definition
 EXTERNAL REFERENCES =
      None.
   DATA AREAS =
      None.
   CONTROL BLOCKS =
      None.
   GLOBAL VARIABLES (Macro pass) =
      None.
==================================================================
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	114	PTE	
(0)	CHARACTER	16	PREFIX	block prefix area
(0)	HALFWORD	2	BLOCK_LENGTH	entry length
(2)	CHARACTER	1	ARROW	'>'
(3)	CHARACTER	3	DFH	'DFH'
(6)	CHARACTER	2	DOMID	'PR'
(8)	CHARACTER	8	BLOCK_NAME	'PTEBLOCK'
(10)	CHARACTER	98	BODY	body of entry
(10)	CHARACTER	8	NAME_PART	name part
(10)	CHARACTER	8	NAME	name of this entry
(18)	CHARACTER	90	ATTRIBUTES_PART	attributes part
(18)	CHARACTER	8	PROFILE_NAME	profile name
(20)	CHARACTER	8	NETWORK	network
(28)	CHARACTER	8	NETNAME	netname
(30)	HALFWORD	2	TP_NAME_LENGTH	TP name length
(32)	CHARACTER	64	TP_NAME	TP name

```
==================================================================
 Structure of a PRM entry in the CICS Global Catalog.
==================================================================
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	CATALOG_ ENTRY_NAME	
(0)	CHARACTER	8	CEN_NAME_PART	
(0)	CHARACTER	8	NAME	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	90	CATALOG_ENTRY	
(0)	CHARACTER	90	CE_ATTR_PART	
(0)	CHARACTER	8	PROFILE_NAME	
(8)	CHARACTER	8	NETWORK	
(10)	CHARACTER	8	NETNAME	
(18)	HALFWORD	2	TP_NAME_LENGTH	
(1A)	CHARACTER	64	TP_NAME	

Constants

Len	Type	Value	Name	Description
8	CHARACTER	PTEBLOCK	PTE_BLOCK_NAMEI	

RDAB Resource definition anchor block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	48	DFHRDAB	RD Anchor Block
(0)	CHARACTER	8	RDAB_HEAD	Set to >DFHRDAB
(8)	ADDRESS	4	TBSS_PTR	Address of DFHTBSS
(C)	ADDRESS	4	TONR_PTR	Address of DFHTONR
(10)	ADDRESS	4	RDAB_RDAL	Ptr to DFHRDAL list
(14)	FULLWORD	4	RDAB_RET_CODE	Ret code for start
(18)	FULLWORD	4	RDAB_SUSPEND_ TOKEN_INIT	Suspend token wait for APRD INIT
(1C)	FULLWORD	4	RDAB_SUSPEND_ TOKEN_RECOVER	Suspend token wait for APRD RECOVER
(20)	ADDRESS	4	RDAB_RDUB	Ptr to RDUB chain
(24)	ADDRESS	4	RDAB_LAST_RDUB	Ptr to end RDUB
(28)	CHARACTER	8	RDAB_SUBPOOL	Subpool token

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	DFHRDAL	RD Action List
(0)	CHARACTER	8	RDAL_HEADER	Set to >DFHRDAL
(8)	FULLWORD	4	RDAL_FORWARD_PTR	RDAL chain ptr
(C)	FULLWORD	4	RDAL_LENGTH	RDAL length
(10)	CHARACTER	2	RDAL_TYPE	'TO' or 'TB'
(12)	CHARACTER	*	RDAL_ELEMENT	RDAL Element

Constants

Len	Type	Value	Name	Description
8	CHARACTER	>DFHRDAB	RDAB_INIT	
8	CHARACTER	>DFHRDAL	RDAL_INIT	

RDUB Resource definition update block

```
CONTROL BLOCK NAME = DFHRDUB
DESCRIPTIVE NAME = CICS Resource definition update Block
SOURCE = DFHRRAB DESIGN part of DFHAPRDR DESIGN
FUNCTION =
    DFHRDUB describes the DSECT for the Resource definition
    Update Block. This block lists deletions that have been
    made by this unit-of-work from tables. It is chained both
    from the RRAB and from the RDAB.
    When an add or quiesce is performed, the contents of RDUBs
    for other tasks are examined to see if we would overwrite
    an entry which may be backed out subsequently.
    If one is found the taskid and tranid are returned as
    though they had been locks found by TMP.
    The Resource Definition Update Block is built by Table
    Builder Services as part of the processing of an Install
    or Delete. It is added both to a chain from the Resource
    definition Recovery Anchor Block (RRAB), and from the
    Resource Definition Anchor Block (RDAB).
    The Resource Definition Update Block is deleted when the
    associated RRAB is deleted.
    Consider the following cases :-
    Task 1 deletes an entry for terminal ABCD
    Task 2 must not be allowed to add another entry for ABCD
    until Task 1 has committed its unit of work. We used to
    use TMP to hold a global lock until Task1's syncpoint but
    this means that we are very limited in the number of install
    requests that can be processed. So now we hold a list of
    update requested TCT names in the RDUB which allows us to
    ensure that full concurrency can occur.
    Another case is that if Task 1 adds an entry for
    WXYZ we must show it to Task 1, but not to Task 2 or 3. For
    tasks which dont specify SHOW_UPDATES on ZGTI this
    happens because TCTTEDAP,TCTTEDDP,TCSEDAP or TCSEDDP are on.
    If SHOW_UPDATES(YES) is specified, ZGTI will INQUIRE_LOCK
    find out if this entry is soft-locked by another task and
    if so, it will not be returned to the requestor.
LIFETIME =
    Created when the first Table Builder call that causes a
    delete is processed.
    Deleted at end of a UOW.
STORAGE CLASS =
    Above 16M line.
LOCATION =
    Chained from the RRAB and the RDAB.
INNER CONTROL BLOCKS =
    None.
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS = None
 MODULE TYPE = Control block definition
EXTERNAL REFERENCES = None
  DATA AREAS = None
  CONTROL BLOCKS = None
  GLOBAL VARIABLES (Macro pass) = None
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	943	DFHRDUB	
(0)	CHARACTER	8	RDUB_HEADER	Set to >DFHRDUB
(8)	ADDRESS	4	RDUB_FWD_ RDAB_PTR	RDAB chain ptr
(C)	ADDRESS	4	RDUB_BWD_ RDAB_PTR	RDAB back-chain ptr
(10)	ADDRESS	4	RDUB_FWD_ RRAB_PTR	RRAB chain ptr
(14)	ADDRESS	4	RDUB_BWD_ RRAB_PTR	RRAB chain ptr
(18)	ADDRESS	4	RDUB_RRAB	RRAB address
(1C)	FULLWORD	4	RDUB_NUMBER	Number of names + 1
(20)	ADDRESS	4	RDUB_DUMMY_PTR	Always zero
(24)	CHARACTER	3	RDUB_TASKI	Task number
(27)	CHARACTER	4	RDUB_TRANI	Transaction Id
(2B)	CHARACTER	18	RDUB_NAMES (50)	Array of names
(2B)	CHARACTER	13	RDUB_LOCK_NAME	Entry name

Offset Hex	Type	Len	Name (Dim)	Description
(38)	CHARACTER	4	RDUB_LOCK_TABLE	Table quiesced
(3C)	BITSTRING	1	RDUB_FLAGS	Flags
	1111		RDUB_LOCK_TYPE	Entry type
	.... 1...		RDUB_LOCK_ QUIESCE	Unquiesce needed?
	.... .1..		RDUB_LOCK_ SHARED	Shared lock
	11		*	Reserved

Constants

Len	Type	Value	Name	Description
8	CHARACTER	>DFHRDUB	RDUB_NAME	
4	DECIMAL	50	RDUB_MAX	

RMDM Recovery manager domain management instance

-

The &dm. Class declaration contains the signatures for the methods and the declaration of the instance data. The instance data structure is the RM Domain anchor block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	280	RMDM	

--
-

This structure is the RM domain global data.

INSTANCE DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	280	INSTANCE_ DATA_BLOCK	
(0)	STRUCTURE IsA(RM_EYE_CATCHER) Protected	16	RMDM_EYE_ CATCHER	Eyecatcher
(0)	UNSIGNED Public	2	RM_EYE_LEN	object length
(2)	UNSIGNED Public	2	RM_EYE_OFFSET	offset of eye-catcher in object
(4)	CHAR Public	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(10)	CHAR Protected	8	RMDM_SUBPOOL	Subpool Token
(18)	ADDRESS Protected	4	RMDM_LOCK_TOKEN	Domain Lock Token
(1C)	OBJECT IsA(RMCLM) Protected	144	RMDM_CLASS_ MANAGER	Class Manager
(1C)	CHAR Protected	144	INSTANCE_ DATA_BLOCK	
(1C)	CHAR Protected	4	NAME (12)	class name
(4C)	ADDRESS Protected	4	INITIALISER (12)	class initialising proc
(7C)	ADDRESS Protected	4	DATA (12)	class data address
(AC)	UNSIGNED Protected	1	RMDM_CURR_ START_TYPE	Current system start type
(AD)	UNSIGNED Protected	1	RMDM_CURR_ START_ALL	Current system start all option
(AE)	FIXED Protected	1	RMDM_CLEAR_ LOG_AT_COLD_START	Clear the log when cold starting
(AF)	UNSIGNED Protected	1	RMDM_CURR_ START_INIT	Initial start
(B0)	CHAR Protected	64	RMDM_PERSISTENT_ DATA	Persistent Data (stored on catalog)

Offset Hex	Type	Len	Name (Dim)	Description
(B0)	CHAR VARY Protected	17	RMDM_LOCAL_LU_NAME	Local LU Name
(C3)	UNSIGNED Protected	1	RMDM_NEXT_START_TYPE	Next Start Type
(C4)	UNSIGNED Protected	1	RMDM_NEXT_START_ALL	Next Start All 0=unset, rmdm_yes/no
(C5)	UNSIGNED Protected	1	RMDM_STATE	Domain State
(C6)	CHAR Protected	8	RMDM_LAST_COLD_TIME	Last time this system was cold started
(CE)	CHAR Protected	8	RMDM_LAST_EMER_TIME	Last time this system was emergency started
(D6)	CHAR Protected	8	RMDM_LAST_ INIT_TIME	Last time this system was initial started
(DE)	CHAR Protected	18	*	padding

Following structure shared with DFHRMUTL utility.
rmdm_auto_override used herein.

Offset Hex	Type	Len	Name (Dim)	Description
(F0)	CHAR Protected	32	RMDM_PERSISTENT_OPTIONS	
(F0)	CHAR Protected	8	RMDM_AUTO_OVERRIDE	AUTOASIS|AUTOCOLD|AUTOINIT AUTODIAG
(F8)	CHAR Protected	8	RMDM_AUTO_OVERRIDE_TIME	STCK when written out
(100)	CHAR Protected	8	RMDM_COLD_COPY_TIME	STCK when COLD_COPY
(108)	BITSTRING Protected	1	RMDM_POPT_ FLAGS	
	1... .... Protected		RMDM_COLD_ COPIED	'1'B =was COLD_COPYed
	.111 1111 Protected		*	padding
(109)	CHAR Protected	7	*	padding
(110)	FIXED Protected	1	RMDM_DIAGNOSTIC_ RUN	global flag
(111)	CHAR Protected	7	*	reserved

--

SHARED DATA

Declared Data

Offset Hex	Type	Len	Name (Dim)
(0)	BITSTRING Public	1	RMDM_LOCK_STATUS
	1... .... Protected		HELD
	.111 1111 Protected		*

Constants

Len	Type	Value	Name	Description
4	DECIMAL	12	RMCLM_MAX_CLASS	Capacity of the class mgr
4	DECIMAL	9	RMDM_NUM_CLASSES	Number of RM classes

RM Classes identified by constant

Len	Type	Value	Name	Description
4	DECIMAL	1	RMCD_CLASSID	
4	DECIMAL	2	RMVP_CLASSID	
4	DECIMAL	3	RMRO_CLASSID	
4	DECIMAL	4	RMUW_CLASSID	
4	DECIMAL	5	RMLK_CLASSID	
4	DECIMAL	6	RMSL_CLASSID	
4	DECIMAL	7	RMNM_CLASSID	
4	DECIMAL	8	RMNS_CLASSID	
4	DECIMAL	9	RMST_CLASSID	

Spare class ids

Len	Type	Value	Name	Description
4	DECIMAL	10	RMDM_CLASSID_ SPARE2	
4	DECIMAL	11	RMDM_CLASSID_ SPARE3	
4	DECIMAL	12	RMDM_CLASSID_ SPARE4	
4	DECIMAL	0	RMDM_LOCK_FREE	
4	DECIMAL	128	RMDM_LOCK_HELD	

lock error codes

Len	Type	Value	Name	Description
4	CHARACTER	ARMA	RMDM_LOCK_ERROR_CODE	
4	CHARACTER	ARMB	RMDM_UNLOCK_ERROR_CODE	

Len	Type	Value	Name	Description
persistent name and persistent types				
8	CHARACTER	DFHRMDM	RMDM_PTYPE	
16	CHARACTER	DFHRMDM_ANCHOR	RMDM_PNAME	
16	CHARACTER	DFHRMDM_OPTIONS	RMDM_POPTIONS_NAME	
persistent auto option values block added				
8	CHARACTER	AUTODIAG	RMDM_OPT_AUTODIAG	
8	CHARACTER	AUTOASIS	RMDM_OPT_AUTOASIS	
8	CHARACTER	AUTOINIT	RMDM_OPT_AUTOINIT	
8	CHARACTER	AUTOCOLD	RMDM_OPT_AUTOCOLD	
8	CHARACTER	AUTOASIS	RMDM_OPT_AUTODFT	
states				
4	DECIMAL	1	RMDM_PRE_ INITIALISING	
4	DECIMAL	2	RMDM_PRE_ INITIALISED	
4	DECIMAL	3	RMDM_INITIALISED	
4	DECIMAL	4	RMDM_QUIESCED	
4	DECIMAL	5	RMDM_TERMINATED	
1	BIT	00000000	LMLM_LOCK_FREE	
1	BIT	10000000	LMLM_LOCK_HELD	

RMID Recovery manager identity instance

-

The rmid class is the Recovery Manager Identity abstract class.

It may only be used by Recovery Manager.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	24	RMID	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	
(8)	CHAR Protected	8	*	
(8)	ADDRESS Protected	4	PREV	
(C)	ADDRESS Protected	4	NEXT	

--
-

The only piece of instance data is the name of the identity.

Declared Data				
(10)	CHAR Protected	4	NAME	

RMLI Recovery manager loggable object identity instance

-

The rmli class is the Recovery Manager Loggable Object Identity class.

It may only be used by Recovery Manager.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	88	RMLI	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	
(8)	CHAR Protected	8	*	
(8)	ADDRESS Protected	4	PREV	
(C)	ADDRESS Protected	4	NEXT	

--
-

The only piece of instance data is the name of the identity.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHAR Protected	4	NAME	

--
-

The instance data, in addition to that inherited from the rmid class, consists of the address of the start delivery, deliver data, end delivery, take keypoint, set chain token, and inquire disjoint chains methods of an instance of (a subclass of) the loggable object class.

Offset Hex	Type	Len	Name (Dim)	Description
Declared Data				
(18)	CHAR Protected	64	INSTANCE_ DATA_BLOCK	RMLI instance data.
(18)	ADDRESS Protected	4	START_DELIVERY	Start delivery method address.
(1C)	ADDRESS Protected	4	DELIVER_DATA	Deliver data method address.
(20)	ADDRESS Protected	4	END_DELIVERY	End delivery method address.
(24)	ADDRESS Protected	4	TAKE_KEYPOINT	Take keypoint method address.
(28)	ADDRESS Protected	4	SET_CHAIN_TOKEN	Set chain token method address.
(2C)	ADDRESS Protected	4	INQUIRE_ DISJOINT_CHAINS	Inquire disjoint chains method address.
(30)	ADDRESS Protected	4	PRE_KEYPOINT	Start Keypoint method address.
(34)	ADDRESS Protected	4	POST_KEYPOINT	Start Keypoint method address.
(38)	CHAR Protected	32	*	reserved for APAR fixes

RMLK Recovery manager link class data

-

This is the declaration for the rmlk_class_data class.

-

The link class data consists of a list of all the links in the system and a tokenset.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	2672	RMLK_CLASS_DATA	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	
Declared Data				
(8)	CHAR Protected	2664	CLASS_DATA_BLOCK	
(8)	STRUCTURE IsA(RM_EYE_CATCHER) Protected	16	EYE_CATCHER	eyecatcher
(8)	UNSIGNED Public	2	RM_EYE_LEN	object length
(A)	UNSIGNED Public	2	RM_EYE_OFFSET	offset of eye-catcher in object
(C)	CHAR Public	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(18)	OBJECT IsA(HOP_DCHAIN) Protected	40	ALL_LINKS_CHAIN	chain of all links in the system
(18)	CHAR Private	4	*	
(20)	CHAR Protected	16	ITER0	
(20)	CHAR Private	4	*	
(28)	CHAR Protected	8	*	
(28)	ADDRESS Protected	4	PREV	
(2C)	ADDRESS Protected	4	NEXT	
(30)	CHAR Protected	16	NODE0	
(30)	CHAR Private	4	*	
(38)	CHAR Protected	8	*	
(38)	ADDRESS Protected	4	PREV	
(3C)	ADDRESS Protected	4	NEXT	
(40)	OBJECT IsA(RMTOKSET) Protected	1056	LINK_TOKENS	

--
-

The token set records the set of known tokens together with the address associated with each known token.

Offset Hex	Type	Len	Name (Dim)	Description
(40)	CHAR Protected	1056	INSTANCE_ DATA_BLOCK	
(40)	CHAR Protected	16	EYE_CATCHER	eyecatcher
(40)	UNSIGNED Public	2	RM_EYE_LEN	object length
(42)	UNSIGNED Public	2	RM_EYE_ OFFSET	offset of eye-catcher in object
(44)	CHAR Public	12	RM_EYE_ STRING	'>DFHRMxxxxxx'
(50)	UNSIGNED Protected	4	NUMBER_ OF_BLOCKS	block count
(54)	UNSIGNED Protected	4	FREE_CHAIN_ HEAD	free chain head
(54)	CHAR Protected	2	INDEX	
(54)	UNSIGNED Protected	1	BLOCK	
(55)	UNSIGNED Protected	1	SLOT	
(56)	UNSIGNED Protected	2	INSTANCE	
(58)	ADDRESS Protected	4	BLOCKS (0 255)	pointers to blocks
(458)	CHAR Protected	8	*	

Offset Hex	Type	Len	Name (Dim)	Description
(460)	OBJECT IsA(RMTOKSET) Protected	1056	BROWSE_TOKENS	token sets
(460)	CHAR Protected	1056	INSTANCE_DATA_BLOCK	
(460)	CHAR Protected	16	EYE_CATCHER	eyecatcher
(460)	UNSIGNED Public	2	RM_EYE_LEN	object length
(462)	UNSIGNED Public	2	RM_EYE_ OFFSET	offset of eye-catcher in object
(464)	CHAR Public	12	RM_EYE_ STRING	'>DFHRMxxxxxx'
(470)	UNSIGNED Protected	4	NUMBER_ OF_BLOCKS	block count
(474)	UNSIGNED Protected	4	FREE_CHAIN_ HEAD	free chain head
(474)	CHAR Protected	2	INDEX	
(474)	UNSIGNED Protected	1	BLOCK	
(475)	UNSIGNED Protected	1	SLOT	
(476)	UNSIGNED Protected	2	INSTANCE	
(478)	ADDRESS Protected	4	BLOCKS (0 255)	pointers to blocks
(878)	CHAR Protected	8	*	
(880)	OBJECT IsA(RMOF) Protected	40	LINK_FACTORY	object factory

-

The instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix 'RMOF' and a suffix which is the name of the object being managed.

(880)	CHAR Protected	40	INSTANCE_DATA_BLOCK	RMOF instance data
(880)	CHAR Protected	16	OF_EYE_ CATCHER	eye-catcher
(880)	UNSIGNED Public	2	RM_EYE_LEN	object length
(882)	UNSIGNED Public	2	RM_EYE_ OFFSET	offset of eye-catcher in object
(884)	CHAR Public	12	RM_EYE_ STRING	'>DFHRMxxxxxx'
(890)	CHAR Protected	8	SUBPOOL_NAME	subpool name
(890)	CHAR Protected	4	SUBPOOL_NAME_PREFIX	subpool name prefix
(894)	CHAR Protected	4	SUBPOOL_NAME_SUFFIX	subpool name suffix
(898)	CHAR Protected	8	SUBPOOL_TOKEN	subpool token
(8A0)	CHAR Protected	8	*	
(8A8)	OBJECT IsA(RMLI) Protected	88	LI	loggable object identity
(8A8)	CHAR Private	4	*	
(8B0)	CHAR Protected	8	*	
(8B0)	ADDRESS Protected	4	PREV	
(8B4)	ADDRESS Protected	4	NEXT	

--
-

The only piece of instance data is the name of the identity.

(8B8)	CHAR Protected	4	NAME	

--
-

The instance data, in addition to that inherited from the rmid class, consists of the address of the start delivery, deliver data, end delivery, take keypoint, set chain token, and inquire disjoint chains methods of an instance of (a subclass of) the loggable object class.

(8C0)	CHAR Protected	64	INSTANCE_DATA_BLOCK	RMLI instance data.
(8C0)	ADDRESS Protected	4	START_DELIVERY	Start delivery method address.

Offset Hex	Type	Len	Name (Dim)	Description
(8C4)	ADDRESS Protected	4	DELIVER_DATA	Deliver data method address.
(8C8)	ADDRESS Protected	4	END_DELIVERY	End delivery method address.
(8CC)	ADDRESS Protected	4	TAKE_KEYPOINT	Take keypoint method address.
(8D0)	ADDRESS Protected	4	SET_CHAIN_ TOKEN	Set chain token method address.
(8D4)	ADDRESS Protected	4	INQUIRE_ DISJOINT_CHAINS	Inquire disjoint chains method address.
(8D8)	ADDRESS Protected	4	PRE_KEYPOINT	Start Keypoint method address.
(8DC)	ADDRESS Protected	4	POST_KEYPOINT	Start Keypoint method address.
(8E0)	CHAR Protected	32	*	
(900)	CHAR Protected	8	LINK_STATISTICS	link-related statistics:
(900)	SIGNED Protected	4	TOTAL_RESYNCS	#resyncs
(904)	SIGNED Protected	4	TOTAL_HEURISTIC_ MISMATCHES	#heuristic mismatches
(908)	OBJECT IsA(RMLK) Protected	288	PROFORMA_LINK	Proforma RMLK object
(908)	CHAR Private	4	*	
(910)	CHAR Protected	276	INSTANCE_ DATA_BLOCK	RMLK Instance Data
(910)	CHAR Protected	16	EYE_CATCHER	eyecatcher
(910)	UNSIGNED Public	2	RM_EYE_LEN	object length
(912)	UNSIGNED Public	2	RM_EYE_ OFFSET	offset of eye-catcher in object
(914)	CHAR Public	12	RM_EYE_ STRING	'>DFHRMxxxxxx'
(920)	CHAR Protected	16	CLASS_CHAIN	chain of all RMLKs in the system
(920)	CHAR Private	4	*	
(928)	CHAR Protected	8	*	
(928)	ADDRESS Protected	4	PREV	
(92C)	ADDRESS Protected	4	NEXT	
(930)	CHAR Protected	16	LINKSET_CHAIN	chain of RMLKs in the same UOW
(930)	CHAR Private	4	*	
(938)	CHAR Protected	8	*	
(938)	ADDRESS Protected	4	PREV	
(93C)	ADDRESS Protected	4	NEXT	
(940)	CHAR Protected	4	LINK_TOKEN	Token of this RMLK
(944)	ADDRESS Protected	4	UOW_POINTER	Address of RMUW
(948)	ADDRESS Protected	4	CLIENT_ POINTER	Address of RMCI
(94C)	ADDRESS Protected	4	UNFORGOTTEN_ LINK_PTR	Address of RMLK that is awaiting forget
(950)	ADDRESS Protected	4	CURRENT_ LINK_PTR	Address of passed RMLK
(954)	BITSTRING Protected	4	LINK_FLAGS	
(954)	BITSTRING Protected	1	*	
	`1... ....` Protected		OWNED_BY_ LINKSET	Not thru syncpoint yet
	`.1.. ....` Protected		CALL_BACK_ IN_PROGRESS	Currently calling client back
	`..1. ....` Protected		UOW_TERMINATE_ RECOVERY_ NECESSARY	Must Terminate_Rec on the UOW
	`...1 ....` Protected		INBOUND_ RECOVERY_ IN_PROGRESS	
	`.... 1...` Protected		OUTBOUND_ RECOVERY_ IN_PROGRESS	

-

Attributes that appear as in CDURUN as enumerated types are held similarly in the object.

Offset Hex	Type	Len	Name (Dim)	Description
	.... .1.. Protected		TO_BE_CLEAR_PENDED	Must be cleared when convenient
	.... ..1. Protected		HAS_BEEN_ISSUE_PREPARED	
	....1 Protected		UOW_SURVIVED_COLD_START	@PKC
(955)	BITSTRING Protected	1	*	
	1... .... Protected		HAS_BEEN_DELETED	
	.1.. .... Protected		PRELOGGING_REQUIRED	
	..11 1111 Protected		*	
(956)	BITSTRING Protected	2	*	
(958)	SIGNED Protected	4	LINK_STATUS	link status
(95C)	CHAR Protected	139	LOGGED_STATE	Data that is logged
(95C)	CHAR Protected	4	CLIENT_NAME	Client name
(960)	ADDRESS Protected	4	RMC_TOKEN	Clients token
(964)	SIGNED Protected	4	TIMES_LOGGED	Number of records for this RMLK on the log
(968)	CHAR Protected	8	FAILURE_TIME	Time when inaccessible
(970)	UNSIGNED Protected	1	PRESUMPTION	
(971)	UNSIGNED Protected	1	COORDINATOR	Other side is coordinator
(972)	UNSIGNED Protected	1	INITIATOR	Other side is intiator
(973)	UNSIGNED Protected	1	LINK_ID_ SOURCE	Which side originated the link id
(974)	UNSIGNED Protected	1	REMOTE_UOW_STATUS	Other sides status
(975)	UNSIGNED Protected	1	FORGET	Whether forgotten
(976)	CHAR VARY Protected	64	LOGNAME	Logname
(9B8)	CHAR VARY Protected	18	LINK_ID	Link id
(9CC)	CHAR VARY Protected	17	ACCESS_ID	Access id
(9DF)	UNSIGNED Protected	1	NO_RESYNC_OUTCOME	No inbound UOW resolution at resync time
(9E0)	CHAR Protected	7	*	
(9E7)	UNSIGNED Protected	1	LAST	Preference for Last Agent
(9E8)	UNSIGNED Protected	1	PRELOGGING	Request for prelogging
(9E9)	UNSIGNED Protected	1	SINGLE_ UPDATER	Supports Single Updater
(9EA)	UNSIGNED Protected	1	RECOVERY_ STATUS	Recovery necessary
(9EB)	UNSIGNED Protected	1	VOTE	
(9EC)	UNSIGNED Protected	1	PASS	RMLK is to be/was passed
(9ED)	UNSIGNED Protected	1	ACCESSIBLE	
(9EE)	UNSIGNED Protected	1	ABEND	Client Abended
(9EF)	UNSIGNED Protected	1	MARK	RMLK marked
(9F0)	UNSIGNED Protected	1	UNSHUNTED	
(9F1)	UNSIGNED Protected	1	RESYNC_ SCHEDULED	
(9F2)	UNSIGNED Protected	1	LOCAL_UOW_ STATUS	
(9F3)	UNSIGNED Protected	1	NEXT_RECOVERY_STATUS	Recovery Status for passed RMLK
(9F4)	UNSIGNED Protected	1	NEXT_SINGLE_UPDATER	Preference for Last Agent for passed RMLK
(9F5)	CHAR Protected	3	*	
(9F8)	SIGNED Protected	4	TIMES_RESTORED	Count of records found on the log
(9FC)	CHAR Protected	40	*	

Constants

Len	Type	Value	Name	Description
4	CHARACTER	RMLK	CLASS_NAME	
4	DECIMAL	0	LINK_RESET	
4	DECIMAL	1	LINK_S_PREPARE	
4	DECIMAL	2	LINK_R_PREPARE	
4	DECIMAL	3	LINK_SELECTED_LAST	
4	DECIMAL	4	LINK_COMMIT	
4	DECIMAL	5	LINK_IN_DOUBT	
4	DECIMAL	6	LINK_S_REQUEST_ COMMIT	
4	DECIMAL	7	LINK_R_REQUEST_ COMMIT	
4	DECIMAL	8	LINK_COMMITTED	
4	DECIMAL	9	LINK_S_COMMITTED	
4	DECIMAL	10	LINK_R_COMMITTED	
4	DECIMAL	11	LINK_R_FORGET	
1	DECIMAL	6	RMLK_MANDATES_LAST	

--

4	DECIMAL	1	RMLK_ABENDED	
4	DECIMAL	2	RMLK_ROLLBACK_ NOT_SUP	

RMLK Recovery manager link instance

-

rmlk is the Recovery Manager Link class.

It may only be used by Recovery Manager. It is used to implement the RMLN gate.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	288	RMLK	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	

-

Attributes that appear as in CDURUN as enumerated types are held similarly in the object.

Offset Hex	Type	Len	Name (Dim)	Description
Declared Data				
(8)	CHAR Protected	276	INSTANCE_ DATA_BLOCK	RMLK Instance Data
(8)	STRUCTURE IsA(RM_EYE_CATCHER) Protected	16	EYE_CATCHER	eyecatcher
(8)	UNSIGNED Public	2	RM_EYE_LEN	object length
(A)	UNSIGNED Public	2	RM_EYE_OFFSET	offset of eye-catcher in object
(C)	CHAR Public	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(18)	OBJECT IsA(HOP_DCHAINNODE) Protected	16	CLASS_CHAIN	chain of all RMLKs in the system
(18)	CHAR Private	4	*	
(20)	CHAR Protected	8	*	
(20)	ADDRESS Protected	4	PREV	
(24)	ADDRESS Protected	4	NEXT	
(28)	OBJECT IsA(HOP_DCHAINNODE) Protected	16	LINKSET_CHAIN	chain of RMLKs in the same UOW
(28)	CHAR Private	4	*	
(30)	CHAR Protected	8	*	
(30)	ADDRESS Protected	4	PREV	
(34)	ADDRESS Protected	4	NEXT	
(38)	STRUCTURE IsA(RM_TOKEN) Protected	4	LINK_TOKEN	Token of this RMLK

Offset Hex	Type	Len	Name (Dim)	Description
(3C)	ADDRESS Protected	4	UOW_POINTER	Address of RMUW
(40)	ADDRESS Protected	4	CLIENT_POINTER	Address of RMCI
(44)	ADDRESS Protected	4	UNFORGOTTEN_LINK_PTR	Address of RMLK that is awaiting forget
(48)	ADDRESS Protected	4	CURRENT_ LINK_PTR	Address of passed RMLK
(4C)	BITSTRING Protected	4	LINK_FLAGS	
(4C)	BITSTRING Protected	1	*	
	1... .... Protected		OWNED_BY_ LINKSET	Not thru syncpoint yet
	.1.. .... Protected		CALL_BACK_IN_PROGRESS	Currently calling client back
	..1. .... Protected		UOW_TERMINATE_RECOVERY_NECESSARY	Must Terminate_Rec on the UOW
	...1 .... Protected		INBOUND_ RECOVERY_IN_PROGRESS	
	.... 1... Protected		OUTBOUND_RECOVERY_IN_PROGRESS	
	.... .1.. Protected		TO_BE_CLEAR_PENDED	Must be cleared when convenient
	.... ..1. Protected		HAS_BEEN_ISSUE_PREPARED	
	....1 Protected		UOW_SURVIVED_COLD_START	@PKC
(4D)	BITSTRING Protected	1	*	
	1... .... Protected		HAS_BEEN_ DELETED	
	.1.. .... Protected		PRELOGGING_REQUIRED	
	..11 1111 Protected		*	
(4E)	BITSTRING Protected	2	*	
(50)	SIGNED Protected	4	LINK_STATUS	link status
(54)	STRUCTURE IsA(RMLK_LOGGED_STATE_TYPE) Protected	139	LOGGED_STATE	Data that is logged
(54)	CHAR Protected	4	CLIENT_NAME	Client name
(58)	ADDRESS Protected	4	RMC_TOKEN	Clients token
(5C)	SIGNED Protected	4	TIMES_LOGGED	Number of records for this RMLK on the log
(60)	CHAR Protected	8	FAILURE_TIME	Time when inaccessible
(68)	UNSIGNED Protected	1	PRESUMPTION	
(69)	UNSIGNED Protected	1	COORDINATOR	Other side is coordinator
(6A)	UNSIGNED Protected	1	INITIATOR	Other side is intiator
(6B)	UNSIGNED Protected	1	LINK_ID_SOURCE	Which side originated the link id
(6C)	UNSIGNED Protected	1	REMOTE_ UOW_STATUS	Other sides status
(6D)	UNSIGNED Protected	1	FORGET	Whether forgotten
(6E)	CHAR VARY Protected	64	LOGNAME	Logname
(B0)	CHAR VARY Protected	18	LINK_ID	Link id
(C4)	CHAR VARY Protected	17	ACCESS_ID	Access id
(D7)	UNSIGNED Protected	1	NO_RESYNC_ OUTCOME	No inbound UOW resolution at resync time
(D8)	CHAR Protected	7	*	
(DF)	UNSIGNED Protected	1	LAST	Preference for Last Agent
(E0)	UNSIGNED Protected	1	PRELOGGING	Request for prelogging
(E1)	UNSIGNED Protected	1	SINGLE_UPDATER	Supports Single Updater
(E2)	UNSIGNED Protected	1	RECOVERY_STATUS	Recovery necessary

Offset Hex	Type	Len	Name (Dim)	Description
(E3)	UNSIGNED Protected	1	VOTE	
(E4)	UNSIGNED Protected	1	PASS	RMLK is to be/was passed
(E5)	UNSIGNED Protected	1	ACCESSIBLE	
(E6)	UNSIGNED Protected	1	ABEND	Client Abended
(E7)	UNSIGNED Protected	1	MARK	RMLK marked
(E8)	UNSIGNED Protected	1	UNSHUNTED	
(E9)	UNSIGNED Protected	1	RESYNC_SCHEDULED	
(EA)	UNSIGNED Protected	1	LOCAL_UOW_STATUS	
(EB)	UNSIGNED Protected	1	NEXT_RECOVERY_STATUS	Recovery Status for passed RMLK
(EC)	UNSIGNED Protected	1	NEXT_SINGLE_UPDATER	Preference for Last Agent for passed RMLK
(ED)	CHAR Protected	3	*	Reserved
(F0)	SIGNED Protected	4	TIMES_RESTORED	Count of records found on the log
(F4)	CHAR Protected	40	*	Reserved
(0)	OBJECT IsA(RMUW) Protected	1360	UOW	
(0)	CHAR Private	4	*	

-

The instance data of a RMUW object includes an instance of a Poller since the inheritance from Poller is simulated.

Offset Hex	Type	Len	Name (Dim)	Description
(8)	CHAR Protected	1352	INSTANCE_DATA_BLOCK	RMUW instance data
(8)	CHAR Protected	16	UOW_EYE_CATCHER	Eye-catcher
(8)	UNSIGNED Public	2	RM_EYE_LEN	object length
(A)	UNSIGNED Public	2	RM_EYE_OFFSET	offset of eye-catcher in object
(C)	CHAR Public	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(18)	CHAR Protected	16	UOW_CHAIN_LINK	Link in global UOW chain
(18)	CHAR Private	4	*	
(20)	CHAR Protected	8	*	
(20)	ADDRESS Protected	4	PREV	
(24)	ADDRESS Protected	4	NEXT	
(28)	CHAR Protected	4	UOW_TOKEN	UOW token
(2C)	UNSIGNED Protected	1	STATUS	UOW status
(2D)	UNSIGNED Protected	1	LINKS_PRESENT	Whether links are left in the UOW
(2E)	UNSIGNED Protected	1	KEYPOINT_COUNT	# of keypoints seen
(2F)	UNSIGNED Protected	1	HEURISTIC_CAUSE	Cause of heurism
(30)	CHAR Protected	3	*	
(33)	CHAR Protected	31	UOW_CONTEXT	context info @POC
(33)	CHAR Protected	20	TRAN_CONTEXT	
(33)	CHAR Public	4	TERMID	Terminal id. of originating transaction
(37)	CHAR Public	8	TERMINAL_LUNAME	Terminal LU name of originating transaction
(3F)	CHAR Public	4	TRANNUM	Transaction number of originating transaction
(43)	CHAR Public	4	TRANID	Transaction id. of originating transaction
(47)	CHAR Protected	8	*	
(47)	CHAR Protected	8	USERID	Userid of originating transaction
(47)	CHAR Protected	8	TRAN_TOKEN	Token for originating transaction
(4F)	CHAR Protected	3	OP_ID	Operator id. of originating transaction
(52)	UNSIGNED Protected	1	HEURISM	Whether to take a heuristic decision on an indoubt failure
(53)	UNSIGNED Protected	1	CHOICE	The default direction for a heuristic decision
(54)	UNSIGNED Protected	4	INDOUBT_TIMEOUT_INTERVAL	Limit of amount of time and indoubt wait will be allowed befor being forced to take a heuristic decision. Zero denotes no time limit.
(58)	BITSTRING Protected	4	FLAGS	Flags.
(58)	BITSTRING Protected	1	*	
	`1... .... Protected`		FIRST_UOW_FOR_TRANSACTION	

Offset Hex	Type	Len	Name (Dim)	Description
				First UOW for a transaction.
	.1.. .... Protected		RECONSTRUCTED	UOW was reconstructed during system restart.
	..1. .... Protected		SHUNTED	UOW is shunted.
	...1 .... Protected		HEURISTIC_ DECISION_TAKEN	A heuristic decision has been taken.
	.... 1... Protected		FORCE_PURGE_ PROTECTION	Protected from force purge.
	.... .1.. Protected		UNSHUNT_ ACTIVE	Unshunt in progress.
	.... ..1. Protected		RESYNCH_ IN_PROGRESS	Resynch. in progress.
	....1 Protected		EXISTENCE_ TO_BE_LOGGED	UOW existence needs logging.
(59)	BITSTRING Protected	1	*	
	1... .... Protected		EXISTENCE_ LOCKED	UOW may not be destroyed yet.
	.1.. .... Protected		RESUME_ REQUIRED	A transaction is suspended on this UOW.
	..1. .... Protected		UNSHUNT_ DEFERRED	Unshunt deferred until later.
	...1 .... Protected		SERIAL_ RECOVERY	UOW is being reconstructed during system restart but its indoubt or inflight log records have not yet been reached.
	.... 1... Protected		MOVE_IN_ PROGRESS	UOW is being moved on the log.
	.... .1.. Protected		LOCALLY_ COMMITTED	local commits done.
	.... ..1. Protected		KEYPOINTED_ FOR_MOVE	keypointed in order to move
	....1 Protected		LINKS_FORGOTTEN	no links left
(5A)	BITSTRING Protected	1	*	
	1... .... Protected		FIRST_COMMIT_ DONE	first attempt at commit completed
	.1.. .... Protected		TIMEOUT_ ACTIVE	Indoubt wait timeout is active for this UOW.
	..1. .... Protected		SURVIVED_ COLD_START	UOW has survived a cold start.
	...1 .... Protected		LOCAL_COMMIT_ LOGGED	logged the fact that UOW has locally comm- itted.
	.... 1... Protected		CLIENT_ STATE_RECOVERED	client state has been recovered
	.... .111 Protected		*	
(5B)	BITSTRING Protected	1	*	
	1... .... Protected		USERID_ FROZEN	userid cannot change
	.111 1111 Protected		*	
(5C)	CHAR Protected	4	SYSTEM_ LOG_CHAIN_TOKEN	System log chain token for this UOW.
(60)	CHAR Protected	8	STATE_CHANGE_ TIME	Time of last change of state
(68)	CHAR Protected	40	UNSHUNT_Q	Queue of unshunt requests.
(68)	CHAR Private	4	*	
(70)	CHAR Protected	16	ITER0	
(70)	CHAR Private	4	*	
(78)	CHAR Protected	8	*	
(78)	ADDRESS Protected	4	PREV	
(7C)	ADDRESS Protected	4	NEXT	
(80)	CHAR Protected	16	NODE0	
(80)	CHAR Private	4	*	
(88)	CHAR Protected	8	*	
(88)	ADDRESS Protected	4	PREV	
(8C)	ADDRESS Protected	4	NEXT	
(90)	UNSIGNED Protected	4	SUSPEND_TOKEN	DS suspend token.
(94)	CHAR Protected	4	*	
(98)	CHAR Protected	32	POLLER	Poller instance.
(98)	CHAR Private	4	*	

Offset Hex	Type	Len	Name (Dim)	Description

--
-

vote is the result of the poll so far.

coordinator is the address of the coordinator voter or zero if there is no coordinator voter.

indoubt determines whether or not we are in the indoubt state. If we are indoubt, then there must be a coordinator voter otherwise there would be no way of resolving the indoubt.

resynchronisation_ in_progress records the resynchronisation state. This prevents multiple concurrent attempt to resynchronise and also protects us from a forced decision during resynchronisation.

read_ only is 'yes' if and only if all the voters polled so far have indicated that they are read-only.

continue is 'yes' if there will be a next UOW. Sometimes there will be a next UOW even when continue is 'no'. This is due to some voter preventing the next UOW from continuing even though the application requested it. In such cases, the next UOW is always aborted without the application having a chance to do further work.

Offset Hex	Type	Len	Name (Dim)	Description
(A0)	CHAR Protected	17	INSTANCE_ DATA_BLOCK	RMPO instance data
(A0)	ADDRESS Protected	4	COORDINATOR	coodinator voter for this poller
(A4)	UNSIGNED Protected	1	VOTE	result of polling so far
(A5)	UNSIGNED Protected	1	INDOUBT	whether or not poller is indoubt
(A6)	UNSIGNED Protected	1	RESYNCHRONISATION_ IN_PROGRESS	whether or not resynch. is in progress
(A7)	UNSIGNED Protected	1	READ_ONLY	read-only result of polling so far
(A8)	UNSIGNED Protected	1	CONTINUE	continuation result of polling so far
(A9)	CHAR Protected	8	*	
(B8)	CHAR Protected	112	LINKS	Set of links from this UOW to remote Recovery Managers.
(B8)	CHAR Private	4	*	

-

A Link Set object contains a chain of all the Links involved in this Unit of Work.

There are embedded Voter and Poller objects and a pointer to the Link picked as last-agent. A Link Set knows whether it is awaiting forget.

Offset Hex	Type	Len	Name (Dim)	Description
(C0)	CHAR Protected	98	INSTANCE_ DATA_BLOCK	
(C0)	CHAR Protected	40	RMLS_LINKS	Chain of link objects
(C0)	CHAR Private	4	*	
(C8)	CHAR Protected	16	ITER0	
(C8)	CHAR Private	4	*	
(D0)	CHAR Protected	8	*	
(D0)	ADDRESS Protected	4	PREV	
(D4)	ADDRESS Protected	4	NEXT	
(D8)	CHAR Protected	16	NODE0	
(D8)	CHAR Private	4	*	
(E0)	CHAR Protected	8	*	
(E0)	ADDRESS Protected	4	PREV	
(E4)	ADDRESS Protected	4	NEXT	
(E8)	ADDRESS Protected	4	RMLS_LAST_ LINK	Pointer to last agent or single updater link
(EC)	CHAR Protected	4	RMLS_VOTER	Voter Object
(EC)	CHAR Private	4	*	
(F0)	CHAR Protected	32	RMLS_POLLER	Poller Object
(F0)	CHAR Private	4	*	
(F8)	CHAR Protected	17	INSTANCE_ DATA_BLOCK	RMPO instance data
(F8)	ADDRESS Protected	4	COORDINATOR	coodinator voter for this poller

Offset Hex	Type	Len	Name (Dim)	Description
(FC)	UNSIGNED Protected	1	VOTE	result of polling so far
(FD)	UNSIGNED Protected	1	INDOUBT	whether or not poller is indoubt
(FE)	UNSIGNED Protected	1	RESYNCHRONISATION_ IN_PROGRESS	whether or not resynch. is in progress
(FF)	UNSIGNED Protected	1	READ_ONLY	read-only result of polling so far
(100)	UNSIGNED Protected	1	CONTINUE	continuation result of polling so far
(101)	CHAR Protected	8	*	
(110)	UNSIGNED Protected	1	RMLS_AWAITING_ FORGET	Linkset is merely awaiting forget
(111)	BITSTRING Protected	1	RMLS_FLAGS	
	`1... .... Protected`		CHAIN_INITIALISED	Chain is initialised
	`.1.. .... Protected`		*	
	`..1. .... Protected`		LINK_COMMIT_ ABENDED	A link abended during perform_commit
	`...1 .... Protected`		LINK_ROLLBACK_ NOT_SUPPORTED	A rollback was tried on a link that does not support it.
(112)	CHAR Protected	8	RMLS_FAILURE_ TIME	Failure time
(11A)	CHAR Protected	8	*	
(128)	CHAR Protected	131	INLINE_ ACCESS_STRUCTURE	Structure of values which may be accessed by inline macro expansions.
(128)	CHAR Protected	8	RMUX_LOCAL_ UOW_ID	
(130)	CHAR Protected	27	RMUX_REMOTE_ UOW_ID	
(130)	UNSIGNED Protected	1	RMUX_REMOTE_ ID_LENGTH	
(131)	UNSIGNED Protected	1	RMUX_REMOTE_ ID_LU_NAME_LENGTH	
(132)	CHAR Protected	25	*	
(14B)	BITSTRING Protected	1	RMUX_FLAGS	
	`1... .... Protected`		OPTIMAL_ CLIENTS_ONLY	
(14C)	ADDRESS Protected	4	RMUX_WORK_ TOKEN_ARRAY (19)	
(198)	CHAR Protected	19	RMUX_CLIENT_ STATES	
(198)	BITSTRING Protected	1	CLIENT_STATE (19)	
	`1... .... Protected`		COMMIT_ COMPLETE	
	`.111 1111 Protected`		*	
(1AB)	CHAR Protected	4	*	
(1B0)	CHAR Protected	48	RO_ARRAY (19)	Resource Owner instances.
(1B0)	CHAR Private	4	*	
(1B8)	CHAR Protected	4	VOTER	
(1B8)	CHAR Private	4	*	

Offset Hex	Type	Len	Name (Dim)	Description

--
-

The RMRO instance is prepared by preparing the corresponding Resource Owner.

--
-

The RMRO instance is committed by committing the corresponding Resource Owner.

--
-

The instance data for a Resource Owner object includes its identity.

A type is declared for force tokens and a null force token is declared.

A log header type is declared the length field of which includes the length of the resource id. which is appended to the header structure. Whether or not there is a resource id. is indicated by the resource id. existence bit. The source field in the discriminant is always 'private' for a resource owner log record as this class is the source of the log record as far as the RM classes are concerned since RM doesn't own or understand the format of data which is passed on the APPEND function.

The backout structure is used during backout and backout retry to track the progress of backout. If the pointer to this structure is null, then either backout has not yet started or else backout has completed successfully. The backout structure itself is declared internally to the class as the users of the class should be insensitive to it.

The commit structure is used for forget processing. If the pointer to this structure is null, then there has been no request forget. The commit structure itself is declared internally to the class as the users of the class should be insensitive to it.

Offset Hex	Type	Len	Name (Dim)	Description
(1C0)	CHAR Protected	28	INSTANCE_ DATA_BLOCK	RMRO instance.
(1C0)	CHAR Protected	4	NAME	Resource Owner client name.
(1C4)	ADDRESS Protected	4	BACKOUT_ STRUCT	Pointer to backout failure structure.
(1C8)	ADDRESS Protected	4	COMMIT_ STRUCT	Pointer to commit failure structure.
(1CC)	ADDRESS Protected	4	CLIENT_ IDENTITY_ADDRESS	Resource Owner client identity address.
(1D0)	BITSTRING Protected	1	SYSTEM_ RESTART_STATES	State during system restart.
	11.. .... Protected		COMMIT_ STATE	Commit state.
	..11 1... Protected		BACKOUT_ STATE	Backout state.
	.... .11. Protected		REQ_FORGET_ STATE	Request forget state.
(1D1)	BITSTRING Protected	1	RO_CLIENT_ FLAGS	
	1... .... Protected		RECORDS_ IGNORED	Records ignored
	.111 1111 Protected		*	
(1D2)	CHAR Protected	10	*	
(540)	CHAR Protected	8	TIMER_TOKEN	TI domain indoubt wait timeout token
(548)	CHAR Protected	8	*	
(0)	OBJECT IsA(RMCI) Protected	136	CLIENT	
(0)	CHAR Private	4	*	
(8)	CHAR Protected	8	*	
(8)	ADDRESS Protected	4	PREV	
(C)	ADDRESS Protected	4	NEXT	

--
-

The only piece of instance data is the name of the identity.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHAR Protected	4	NAME	

Offset Hex	Type	Len	Name (Dim)	Description

--
-

As &ci. class is a subclass of &id. class each &ci. is an &id..

Each &ci. also records the client type, the domain and gate for calls back to the named client. They also have a chain representing tasks waiting to call back a client that has not yet set its gate. The objects on the waiters chain are contained in the automatic storage of the waiting task.

The Send method allows one call to be made to the client before the gate is set without suspending the calling task. In this case the parameter list being sent to the client is copied and hung off the &ci. by rmci_ sent_plist_ptr.

(18)	CHAR Protected	112	INSTANCE_ DATA_BLOCK	
(18)	CHAR Protected	24	RMCI_PCHAINNODE	Persistent Chain Node

-

An instance of this class consists of a persistent name and a boolean to indicate whether or not the object has been recovered or not.

(18)	CHAR Protected	16	INSTANCE_ DATA_BLOCK	
(18)	CHAR Protected	8	PERSISTENT_ NAME	persistent name
(20)	BITSTRING Protected	1	FLAGS	
	1... .... Protected		RECOVERED	Is the object recovered?
	.111 1111 Protected		*	
(21)	CHAR Protected	7	*	

-

Each Persistent Node points to the Persistent Collection it belongs to. The Persistent Collection is the Persistent Store for the Persistent Node.

(28)	ADDRESS Protected	4	STORE_POINTER	
(30)	UNSIGNED Protected	1	RMCI_REGISTERED	Has the client registered?
(31)	UNSIGNED Protected	1	RMCI_TYPE	Client type
(32)	CHAR Protected	2	*	
(34)	UNSIGNED Protected	4	RMCI_DOMAIN	Client Domain
(38)	UNSIGNED Protected	4	RMCI_GATE	Client Callback Gate
(3C)	CHAR Protected	4	*	
(40)	CHAR Protected	40	RMCI_WAITERS	Chain of tasks waiting to call the client after the gate has been set
(40)	CHAR Private	4	*	
(48)	CHAR Protected	16	ITER0	
(48)	CHAR Private	4	*	
(50)	CHAR Protected	8	*	
(50)	ADDRESS Protected	4	PREV	
(54)	ADDRESS Protected	4	NEXT	
(58)	CHAR Protected	16	NODE0	
(58)	CHAR Private	4	*	
(60)	CHAR Protected	8	*	
(60)	ADDRESS Protected	4	PREV	
(64)	ADDRESS Protected	4	NEXT	
(68)	ADDRESS Protected	4	RMCI_SENT_ PLIST_PTR	Pointer to the parameter list to being sent
(6C)	ADDRESS Protected	4	RMCI_RMNS_PTR	Pointer to the set of log- names known to this client
(70)	ADDRESS Protected	4	RMCI_PERSISTENT_ DATA_PTR	Pointer to the clients persistent data
(74)	CHAR Protected	20	*	
(0)	CHAR Protected	66	RMCI_PERSISTENT_ DATA	
(0)	CHAR VARY Protected	64	RMCI_CLIENT_ DATA	

SHARED DATA
Declared Data

(0)	CHAR VARY Protected	18	LINK_ID_TYPE	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	139	RMLK_LOGGED_ STATE_TYPE	RMLK as it appears on the log
(0)	CHAR Protected	4	CLIENT_NAME	Client name
(4)	ADDRESS Protected	4	RMC_TOKEN	Clients token
(8)	SIGNED Protected	4	TIMES_LOGGED	Number of records for this RMLK on the log
(C)	CHAR Protected	8	FAILURE_TIME	Time when inaccessible
(14)	UNSIGNED Protected	1	PRESUMPTION	
(15)	UNSIGNED Protected	1	COORDINATOR	Other side is coordinator
(16)	UNSIGNED Protected	1	INITIATOR	Other side is intiator
(17)	UNSIGNED Protected	1	LINK_ID_SOURCE	Which side originated the link id
(18)	UNSIGNED Protected	1	REMOTE_ UOW_STATUS	Other sides status
(19)	UNSIGNED Protected	1	FORGET	Whether forgotten
(1A)	CHAR VARY Protected	64	LOGNAME	Logname
(5C)	CHAR VARY IsA(LINK_ID_TYPE) Protected	18	LINK_ID	Link id
(70)	CHAR VARY Protected	17	ACCESS_ID	Access id
(83)	UNSIGNED Protected	1	NO_RESYNC_ OUTCOME	No inbound UOW resolution at resync time
(84)	CHAR Protected	7	*	
(0)	CHAR Public	139	RMLK_LOGGED_TYPE	

Constants

Len	Type	Value	Name	Description
4	CHARACTER	RMLK	CLASS_NAME	
4	DECIMAL	0	LINK_RESET	
4	DECIMAL	1	LINK_S_PREPARE	
4	DECIMAL	2	LINK_R_PREPARE	
4	DECIMAL	3	LINK_SELECTED_LAST	
4	DECIMAL	4	LINK_COMMIT	
4	DECIMAL	5	LINK_IN_DOUBT	
4	DECIMAL	6	LINK_S_REQUEST_ COMMIT	
4	DECIMAL	7	LINK_R_REQUEST_ COMMIT	
4	DECIMAL	8	LINK_COMMITTED	
4	DECIMAL	9	LINK_S_COMMITTED	
4	DECIMAL	10	LINK_R_COMMITTED	
4	DECIMAL	11	LINK_R_FORGET	
1	DECIMAL	6	RMLK_MANDATES_LAST	
--				
4	DECIMAL	1	RMLK_ABENDED	
4	DECIMAL	2	RMLK_ROLLBACK_ NOT_SUP	

RMLS Recovery manager link set instance

-

This is the class declaration for the Recovery Manager LinkSet class

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	112	RMLS	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	

-

A Link Set object contains a chain of all the Links involved in this Unit of Work.

There are embedded Voter and Poller objects and a pointer to the Link picked as last-agent. A Link Set knows whether it is awaiting forget.

Offset Hex	Type	Len	Name (Dim)	Description
Declared Data				
(8)	CHAR Protected	98	INSTANCE_ DATA_BLOCK	
(8)	OBJECT IsA(HOP_DCHAIN) Protected	40	RMLS_LINKS	Chain of link objects
(8)	CHAR Private	4	*	
(10)	CHAR Protected	16	ITER0	
(10)	CHAR Private	4	*	
(18)	CHAR Protected	8	*	
(18)	ADDRESS Protected	4	PREV	
(1C)	ADDRESS Protected	4	NEXT	
(20)	CHAR Protected	16	NODE0	
(20)	CHAR Private	4	*	
(28)	CHAR Protected	8	*	
(28)	ADDRESS Protected	4	PREV	
(2C)	ADDRESS Protected	4	NEXT	
(30)	ADDRESS Protected	4	RMLS_LAST_LINK	Pointer to last agent or single updater link
(34)	OBJECT IsA(RMVO) Protected	4	RMLS_VOTER	Voter Object
(34)	CHAR Private	4	*	
(38)	OBJECT IsA(RMPO) Protected	32	RMLS_POLLER	Poller Object
(38)	CHAR Private	4	*	

--
-

vote is the result of the poll so far.

coordinator is the address of the coordinator voter or zero if there is no coordinator voter.

indoubt determines whether or not we are in the indoubt state. If we are indoubt, then there must be a coordinator voter otherwise there would be no way of resolving the indoubt.

resynchronisation_ in_progress records the resynchronisation state. This prevents multiple concurrent attempt to resynchronise and also protects us from a forced decision during resynchronisation.

read_ only is 'yes' if and only if all the voters polled so far have indicated that they are read-only.

continue is 'yes' if there will be a next UOW. Sometimes there will be a next UOW even when continue is 'no'. This is due to some voter preventing the next UOW from continuing even though the application requested it. In such cases, the next UOW is always aborted without the application having a chance to do further work.

Offset Hex	Type	Len	Name (Dim)	Description
(40)	CHAR Protected	17	INSTANCE_ DATA_BLOCK	RMPO instance data
(40)	ADDRESS Protected	4	COORDINATOR	coodinator voter for this poller
(44)	UNSIGNED Protected	1	VOTE	result of polling so far
(45)	UNSIGNED Protected	1	INDOUBT	whether or not poller is indoubt
(46)	UNSIGNED Protected	1	RESYNCHRONISATION_ IN_PROGRESS	whether or not resynch. is in progress
(47)	UNSIGNED Protected	1	READ_ONLY	read-only result of polling so far
(48)	UNSIGNED Protected	1	CONTINUE	continuation result of polling so far
(49)	CHAR Protected	8	*	
(58)	FIXED Protected	1	RMLS_AWAITING_ FORGET	Linkset is merely awaiting forget
(59)	BITSTRING Protected	1	RMLS_FLAGS	
	`1... .... Protected`		CHAIN_INITIALISED	Chain is initialised
	`.1.. .... Protected`		*	Reserved
	`..1. .... Protected`		LINK_COMMIT_ ABENDED	A link abended during perform_commit
	`...1 .... Protected`		LINK_ROLLBACK_ NOT_SUPPORTED	A rollback was tried on a link that does not support it.
(5A)	CHAR Protected	8	RMLS_FAILURE_ TIME	Failure time
(62)	CHAR Protected	8	*	Reserved

Constants

Len	Type	Value	Name	Description
4	DECIMAL	1	RMLS_ABENDED	
4	DECIMAL	2	RMLS_ROLLBACK_ NOT_SUPPORTED	
4	DECIMAL	3	RMLS_LINKS_INVALID	

RMNM Recovery manager logname class data

-

This declares the RMNM_class_data class.

-

This structure defines the class data for the &nm. class.

The &nm. class manages the local logname. This is persistent data so there is a &ps. to store it in and a persistent name for it to be known by.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	112	RMNM_CLASS_DATA	
INSTANCE DATA				
Declared Data				
(0)	CHAR Protected	106	CLASS_DATA_BLOCK	
(0)	STRUCTURE IsA(RM_EYE_CATCHER) Protected	16	RMNM_EYE_ CATCHER	eyecatcher
(0)	UNSIGNED Public	2	RM_EYE_LEN	object length
(2)	UNSIGNED Public	2	RM_EYE_OFFSET	offset of eye-catcher in object
(4)	CHAR Public	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(10)	CHAR Protected	74	RMNM_PERSISTENT_ DATA	persistent data
(10)	CHAR VARY Protected	64	RMNM_LOCAL_ LOGNAME	the local logname
(52)	CHAR Protected	8	RMNM_LOCAL_ APPLID	the applid that goes with the log name
(5A)	OBJECT IsA(RMPS) Protected	8	RMNM_PSTORE	persistent store
(5A)	CHAR Protected	8	NAME	
(62)	CHAR Protected	8	*	reserved

Constants

Len	Type	Value	Name	Description
16	CHARACTER	DFHRMNMCLASSDATA	RMNM_CLASS_PNAME	

RMNM Recovery manager logname instance

-

This copybook contains both the RMNM Class and RMNS Class declarations.

-

The &nm. class inherits from the &dn. class so that instances can be collected into &dc.s.

The RMNM Class declaration contains

- the public types used in the interface to the class,

- the instance and class data of the class

- the the signatures of the methods provided by the class and

- the implementations of the internal, inlineable methods.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	168	RMNM	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	
(8)	CHAR Protected	8	*	
(8)	ADDRESS Protected	4	PREV	
(C)	ADDRESS Protected	4	NEXT	

--
-

An instance of this class consists of

- a triple of access_ id, logname and rmc_ data,

- an instance of the Persistent Node class to support persistence.

Declared Data				
(10)	CHAR Protected	152	INSTANCE_ DATA_BLOCK	
(10)	CHAR Protected	119	PERSISTENT_DATA	persistent data
(10)	CHAR VARY Protected	17	ACCESS_ID	access id
(23)	CHAR VARY Protected	64	LOGNAME	logname
(65)	CHAR VARY IsA(RMNM_RMC_DATA_TYPE) Protected	32	RMC_DATA	data held on behalf of the RMC
(87)	CHAR Protected	1	*	reserved
(88)	OBJECT IsA(RMPN) Protected	24	PCHAINNODE	a node in a persistent chain

-

An instance of this class consists of a persistent name and a boolean to indicate whether or not the object has been recovered or not.

(88)	CHAR Protected	16	INSTANCE_ DATA_BLOCK	
(88)	CHAR Protected	8	PERSISTENT_ NAME	persistent name
(90)	BITSTRING Protected	1	FLAGS	
	`1... .... Protected`		RECOVERED	Is the object recovered?
	`.111 1111 Protected`		*	
(91)	CHAR Protected	7	*	

-

Each Persistent Node points to the Persistent Collection it belongs to. The Persistent Collection is the Persistent Store for the Persistent Node.

Offset Hex	Type	Len	Name (Dim)	Description
(98)	ADDRESS Protected	4	STORE_POINTER	
(A0)	CHAR Protected	8	*	

-

The Log Names class deals with data as varying length character strings.

There is also a public type to describe the storage occupied by a flattened version of an instance.

Offset Hex	Type	Len	Name (Dim)	Description
SHARED DATA				
Declared Data				
(0)	CHAR VARY Public	32	RMNM_RMC_ DATA_TYPE	
(0)	CHAR Protected	119	RMNM_FLAT_TYPE	

Constants

Len	Type	Value	Name	Description
16	CHARACTER	DFHRMNMCLASSDATA	RMNM_CLASS_PNAME	

RMNS Recovery manager logname set instance

-

This declares the Recovery Manager RMNS class.

-

The RMNS Class declaration contains

- the instance and class data of the class
- the the signatures of the methods provided by the class and
- the implementations of the internal, inlineable methods.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	72	RMNS	
INSTANCE DATA				
Declared Data				
(0)	OBJECT IsA(HOP_DCHAINNODE) Protected	16	DCHAINNODE	
Inherited Data				
(0)	CHAR Private	4	*	
(8)	CHAR Protected	8	*	
(8)	ADDRESS Protected	4	PREV	
(C)	ADDRESS Protected	4	NEXT	
(10)	OBJECT IsA(HOP_DCHAIN) Protected	40	DCHAIN	
(10)	CHAR Private	4	*	
(18)	CHAR Protected	16	ITER0	
(18)	CHAR Private	4	*	
(20)	CHAR Protected	8	*	
(20)	ADDRESS Protected	4	PREV	
(24)	ADDRESS Protected	4	NEXT	
(28)	CHAR Protected	16	NODE0	
(28)	CHAR Private	4	*	
(30)	CHAR Protected	8	*	
(30)	ADDRESS Protected	4	PREV	
(34)	ADDRESS Protected	4	NEXT	

Offset Hex	Type	Len	Name (Dim)	Description
(38)	OBJECT IsA(RMPC) Protected	16	PCHAIN	

-

An instance of this class consists of a persistent name and a boolean to indicate whether or not the object has been recovered or not.

Offset Hex	Type	Len	Name (Dim)	Description
(38)	CHAR Protected	16	INSTANCE_ DATA_BLOCK	
(38)	CHAR Protected	8	PERSISTENT_ NAME	persistent name
(40)	BITSTRING Protected	1	FLAGS	
	1... .... Protected		RECOVERED	Is the object recovered?
	.111 1111 Protected		*	
(41)	CHAR Protected	7	*	

-

An instance of this class consists of

- a HOP_ Dchain collecting the Log Names objects,

- a Persistent Collection collecting the Persistent Node objects with each Log Names object

- a HOP_ DChainNode to allow the instance to be collected on the HOP_ DChain of known Log Name Set objects maintained by the class.

SHARED DATA

Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	16	RMNS_RECORD_ NAME_TYPE	
(0)	STRUCTURE IsA(RMPE_NAME_TYPE) Protected	8	RMNS_INSTANCE	
(8)	STRUCTURE IsA(RMPE_NAME_TYPE) Protected	8	RMNM_INSTANCE	

--
-

The class data of this class consists of

- an eyecatcher,

- an instance of the Persistent Store class,

- a HOP_ DChain to collect known instances of the class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	64	CLASS_DATA	
(0)	STRUCTURE IsA(RM_EYE_CATCHER) Protected	16	EYE_CATCHER	
(0)	UNSIGNED Public	2	RM_EYE_LEN	object length
(2)	UNSIGNED Public	2	RM_EYE_OFFSET	offset of eye-catcher in object
(4)	CHAR Public	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(10)	OBJECT IsA(RMPS) Protected	8	PSTORE	
(10)	CHAR Protected	8	NAME	
(18)	OBJECT IsA(HOP_DCHAIN) Protected	40	KNOWN_INSTANCES	

Inherited Data

Offset Hex	Type	Len	Name (Dim)	Description
(18)	CHAR Private	4	*	
(20)	CHAR Protected	16	ITER0	
(20)	CHAR Private	4	*	
(28)	CHAR Protected	8	*	
(28)	ADDRESS Protected	4	PREV	
(2C)	ADDRESS Protected	4	NEXT	
(30)	CHAR Protected	16	NODE0	
(30)	CHAR Private	4	*	
(38)	CHAR Protected	8	*	
(38)	ADDRESS Protected	4	PREV	
(3C)	ADDRESS Protected	4	NEXT	

Constants

Len	Type	Value	Name	Description
16	CHARACTER	DFHRMNMCLASSDATA	RMNM_CLASS_PNAME	

RMRO Recovery manager resource owner instance

-

All classes in &rm. domain inherit from the &rm. Object Class (RMOB). This class is completely virtual and contains no data, either class or instance. It merely provides signatures for commom methods that all &rm. domain classes may need. As virtual methods, it is the responsibility of a concrete class inheriting from RMOB to provide implementations of these methods.

Currently there are two such methods. Both are class methods (they don't take an object of the class as a parameter).

--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	48	RMRO	,rmvo
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	
Declared Data				
(8)	OBJECT IsA(RMVO) Protected	4	VOTER	
(8)	CHAR Private	4	*	

--
-

The RMRO instance is prepared by preparing the corresponding Resource Owner.

--
-

The RMRO instance is committed by committing the corresponding Resource Owner.

--
-

The instance data for a Resource Owner object includes its identity.

A type is declared for force tokens and a null force token is declared.

A log header type is declared the length field of which includes the length of the resource id. which is appended to the header structure. Whether or not there is a resource id. is indicated by the resource id. existence bit. The source field in the discriminant is always 'private' for a resource owner log record as this class is the source of the log record as far as the RM classes are concerned since RM doesn't own or understand the format of data which is passed on the APPEND function.

The backout structure is used during backout and backout retry to track the progress of backout. If the pointer to this structure is null, then either backout has not yet started or else backout has completed successfully. The backout structure itself is declared internally to the class as the users of the class should be insensitive to it.

The commit structure is used for forget processing. If the pointer to this structure is null, then there has been no request forget. The commit structure itself is declared internally to the class as the users of the class should be insensitive to it.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHAR Protected	28	INSTANCE_ DATA_BLOCK	RMRO instance.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHAR Protected	4	NAME	Resource Owner client name.
(14)	ADDRESS Protected	4	BACKOUT_STRUCT	Pointer to backout failure structure.
(18)	ADDRESS Protected	4	COMMIT_STRUCT	Pointer to commit failure structure.
(1C)	ADDRESS Protected	4	CLIENT_IDENTITY_ADDRESS	Resource Owner client identity address.
(20)	BITSTRING Protected	1	SYSTEM_RESTART_STATES	State during system restart.
	`11.. .... Protected`		COMMIT_STATE	Commit state.
	`..11 1... Protected`		BACKOUT_STATE	Backout state.
	`.... .11. Protected`		REQ_FORGET_ STATE	Request forget state.
(21)	BITSTRING Protected	1	RO_CLIENT_FLAGS	
	`1... .... Protected`		RECORDS_ IGNORED	Records ignored
	`.111 1111 Protected`		*	Reserved
(22)	CHAR Protected	10	*	reserved for APAR fixes
SHARED DATA				
Declared Data				
(0)	FIXED Public	4	RMRO_FORCE_TOKEN	
(0)	FIXED Protected	1	RMRO_LOG_RECORD_TYPE	
(0)	CHAR Protected	11	RMRO_CD_LOG_HDR	
(0)	STRUCTURE IsA(RMLG_DISCRIMINANT) Protected	7	RMRO_CDLH_DISCRIMINANT	
(0)	UNSIGNED Public	2	RMLG_HEADER_LENGTH	
(2)	CHAR Public	1	RMLG_SOURCE	
(3)	CHAR Public	4	RMLG_NAME	
(7)	FIXED Protected	1	RMRO_CDLH_TYPE	
(8)	BITSTRING Protected	1	RMRO_CDLH_FLAGS	
	`1... .... Protected`		RMRO_CDLH_FORWARD_DATA	
	`.1.. .... Protected`		RMRO_CDLH_BACKWARD_DATA	
	`..1. .... Protected`		RMRO_CDLH_RESOURCE_ID_X	
	`...1 .... Protected`		RMRO_CDLH_FORGET_REQUESTED	
(9)	UNSIGNED Protected	2	RMRO_CDLH_RESOURCE_ ID_LENGTH	
(B)	CHAR Protected		RMRO_CDLH_RESOURCE_ID	
(0)	CHAR Protected	8	RMRO_BFAIL_ LOG_HDR	
(0)	STRUCTURE IsA(RMLG_DISCRIMINANT) Protected	7	RMRO_BFAILLH_DISCRIMINANT	
(0)	UNSIGNED Public	2	RMLG_HEADER_LENGTH	
(2)	CHAR Public	1	RMLG_SOURCE	
(3)	CHAR Public	4	RMLG_NAME	
(7)	FIXED Protected	1	RMRO_BFAILLH_ TYPE	
(0)	CHAR Protected	18	RMRO_BFAIL_MEMBER_LOG_HDR	
(0)	STRUCTURE IsA(RMLG_DISCRIMINANT) Protected	7	RMRO_BFAILMEMLH_DISCRIMINANT	
(0)	UNSIGNED Public	2	RMLG_HEADER_LENGTH	
(2)	CHAR Public	1	RMLG_SOURCE	
(3)	CHAR Public	4	RMLG_NAME	
(7)	FIXED Protected	1	RMRO_BFAILMEMLH_TYPE	
(8)	CHAR VARY Protected	8	RMRO_BFAILMEMLH_RESOURCE_ID	
(12)	CHAR Protected		RMRO_BFAILMEMLH_LOCAL_ACCESS_ID	
(0)	CHAR Protected	10	RMRO_REQ_FORGET_LOG_HDR	
(0)	STRUCTURE IsA(RMLG_DISCRIMINANT) Protected	7	RMRO_RF_DISCRIMINANT	
(0)	UNSIGNED Public	2	RMLG_HEADER_LENGTH	
(2)	CHAR Public	1	RMLG_SOURCE	
(3)	CHAR Public	4	RMLG_NAME	
(7)	FIXED Protected	1	RMRO_RF_TYPE	
(8)	UNSIGNED Protected	2	RMRO_RF_LOCAL_ACCESS_ ID_LEN	

Offset Hex	Type	Len	Name (Dim)	Description
(A)	CHAR Protected		RMRO_RF_LOCAL_ACCESS_ID	
(0)	CHAR Protected	8	RMRO_FORGOTTEN_LOG_HDR	
(0)	STRUCTURE IsA(RMLG_DISCRIMINANT) Protected	7	RMRO_FO_DISCRIMINANT	
(0)	UNSIGNED Public	2	RMLG_HEADER_LENGTH	
(2)	CHAR Public	1	RMLG_SOURCE	
(3)	CHAR Public	4	RMLG_NAME	
(7)	UNSIGNED Protected	1	RMRO_FO_TYPE	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	0	NULL_RMRO_FORCE_TOKEN	
1	DECIMAL	1	RMRO_TYPE_ CLIENT_DATA	
1	DECIMAL	2	RMRO_TYPE_ BFAIL_BEGIN	
1	DECIMAL	3	RMRO_TYPE_BFAIL_MEMBER	
1	DECIMAL	4	RMRO_TYPE_ BFAIL_END	
1	DECIMAL	5	RMRO_TYPE_ REQ_FORGET	
1	DECIMAL	6	RMRO_TYPE_ FORGOTTEN	
0	BIT	00	CS_RESET	
0	BIT	01	CS_COMMIT_COMPLETE	
0	BIT	10	CS_BUILDING_TBF	
0	BIT	11	CS_COMMIT_FAILED	
0	BIT	000	BS_RESET	
0	BIT	001	BS_NOT_BACKED_OUT	
0	BIT	010	BS_BACKOUT_ COMPLETE	
0	BIT	011	BS_BACKOUT_FAILED	
0	BIT	100	BS_REBUILDING_ FAILURE	
0	BIT	00	RF_RESET	
0	BIT	01	RF_FORGOTTEN	
0	BIT	10	RF_FORGET_REQUIRED	

-

The class data consists of the identity object for system logging. Its purpose is to allow the delivery method to distinguish records which are being delivered from RMSL from those which are being delivered from RMUW. In most cases, the content of the records is sufficient to make this distinction, but using different identities (i.e. with different scope values) for system and UOW logging is more general and allows identical log records to be logged to RMSL and RMUW without risk of confusion on delivery.

4	CHARACTER	RMRO	RMRO_SYSTEM_LOG_ID_NAME	
4	CHARACTER		RMRO_SPARE_NAME	

RMSL Recovery manager system log instance

-

The rmsl class is the Recovery Manager System Log.

It may only be used by Recovery Manager.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	128	RMSL	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	

--

Offset Hex	Type	Len	Name (Dim)	Description
Declared Data				
(8)	CHAR Protected	117	INSTANCE_ DATA_BLOCK	RMSL instance.
(8)	STRUCTURE IsA(RM_EYE_CATCHER) Protected	16	RMSL_EYE_ CATCHER	Eye-catcher.
(8)	UNSIGNED Public	2	RM_EYE_LEN	object length
(A)	UNSIGNED Public	2	RM_EYE_OFFSET	offset of eye-catcher in object
(C)	CHAR Public	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(18)	STRUCTURE IsA(RESTART_STATE_TYPE) Protected	4	RESTART_STATE	System restart state of RMSL.
(1C)	FIXED Protected	1	KEYPOINT_ SCHEDULED	Whether or not a keypoint is scheduled.
(1D)	FIXED Protected	1	QUIESCE_ IN_PROGRESS	Whether or not a system quiesce is in progress.
(1E)	FIXED Protected	1	WARM_KP_ WAITING_ FOR_AKP_END	Whether or a warm keypoint is waiting for an activity keypoint to complete before proceeding.
(1F)	STRUCTURE IsA(RMSL_CHAIN) Protected	4	KEYPOINT_CHAIN	System log chain token used for a keypoint.
(23)	FIXED Protected	1	CHAIN_CLOSED	Whether or not a chain has been closed.
(24)	CHAR Protected	4	*	Reserved
(28)	OBJECT IsA(RMCR) Protected	40	SYSTEM_ LOG_REGISTER	Register of clients of RMSL.

--
-

A Client Register is just a chain of Identitys.

Offset Hex	Type	Len	Name (Dim)	Description
(28)	CHAR Protected	40	RMCR_CHAIN	
(28)	CHAR Private	4	*	
(30)	CHAR Protected	16	ITER0	
(30)	CHAR Private	4	*	
(38)	CHAR Protected	8	*	
(38)	ADDRESS Protected	4	PREV	
(3C)	ADDRESS Protected	4	NEXT	
(40)	CHAR Protected	16	NODE0	
(40)	CHAR Private	4	*	
(48)	CHAR Protected	8	*	
(48)	ADDRESS Protected	4	PREV	
(4C)	ADDRESS Protected	4	NEXT	
(50)	STRUCTURE IsA(RMSL_CHAIN) Protected	4	COLD_START_ CHAIN	System log chain token used for cold start.
(54)	FIXED Protected	1	IN_COLD_STATE	Currently in cold start log records
(55)	CHAR Protected	40	*	reserved for APAR fixes
SHARED DATA				
Declared Data				
(0)	CHAR Public	4	RESTART_ STATE_TYPE	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Public	4	RMSL_CHAIN	
(0)	CHAR Protected	28	RMSL_LOG_HEADER	
(0)	STRUCTURE IsA(RMLG_DISCRIMINANT) Protected	7	RMSL_LH_ DISCRIMINANT	
(0)	UNSIGNED Public	2	RMLG_HEADER_LENGTH	
(2)	CHAR Public	1	RMLG_SOURCE	
(3)	CHAR Public	4	RMLG_NAME	
(7)	BITSTRING Protected	1	RMSL_LH_FLAGS	
	`1... .... Protected`		RMSL_LH_ KEYPOINT	
	`.1.. .... Protected`		RMSL_LH_START_OF_KEYPOINT	
	`..1. .... Protected`		RMSL_LH_END_OF_KEYPOINT	
	`...1 .... Protected`		RMSL_LH_ START_OF_COLD_RECOVERY	
	`.... 1... Protected`		RMSL_LH_ END_OF_COLD_RECOVERY	
(8)	CHAR Protected	4	RMSL_LH_TERMID	
(C)	CHAR Protected	8	RMSL_LH_TERMINAL_LUNAME	
(14)	CHAR Protected	4	RMSL_LH_TRANID	
(18)	CHAR Protected	4	RMSL_LH_TASKID	
(1C)	CHAR Protected		RMSL_LH_DATA	

Constants

Len	Type	Value	Name	Description
4	CHARACTER	Init	RS_RESET	
4	CHARACTER	Cold	RS_COLD	
4	CHARACTER	DelP	RS_DELIVERY_IN_PROGRESS	
4	CHARACTER	InKP	RS_KEYPOINT_IN_PROGRESS	
4	CHARACTER	PreK	RS_PRE_KEYPOINT	
4	CHARACTER	Disj	RS_DISJOINT	
4	CHARACTER	KPDe	RS_KEYPOINT_ DELIVERY	
4	CHARACTER	Done	RS_COMPLETE	
4	CHARACTER		RMSL_NULL_CHAIN	
4	DECIMAL	1	RMSL_BUFFER_FULL	
4	DECIMAL	2	RMSL_INVALID_DATA_LENGTH	

RMSL Recovery manager system log class data

-

This declares the Recovery Manager System Log Class Data class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	128	RMSL_CLASS_DATA	

-

The class data just contains the single rmsl instance. The name 'solitaire' reflects the design pattern which is being used.

INSTANCE DATA				
Declared Data				
(0)	OBJECT IsA(RMSL) Protected	128	SOLITAIRE_ SYSTEM_LOG	
Inherited Data				
(0)	CHAR Private	4	*	

--

(8)	CHAR Protected	117	INSTANCE_ DATA_BLOCK	RMSL instance.
(8)	CHAR Protected	16	RMSL_EYE_ CATCHER	Eye-catcher.
(8)	UNSIGNED Public	2	RM_EYE_LEN	object length
(A)	UNSIGNED Public	2	RM_EYE_OFFSET	offset of eye-catcher in object
(C)	CHAR Public	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(18)	CHAR Protected	4	RESTART_STATE	System restart state of RMSL.
(1C)	UNSIGNED Protected	1	KEYPOINT_ SCHEDULED	Whether or not a keypoint is scheduled.
(1D)	UNSIGNED Protected	1	QUIESCE_ IN_PROGRESS	Whether or not a system quiesce is in progress.
(1E)	UNSIGNED Protected	1	WARM_KP_ WAITING_ FOR_AKP_END	Whether or a warm keypoint is waiting for an activity keypoint to complete before proceeding.
(1F)	CHAR Protected	4	KEYPOINT_CHAIN	System log chain token used for a keypoint.
(23)	UNSIGNED Protected	1	CHAIN_CLOSED	Whether or not a chain has been closed.
(24)	CHAR Protected	4	*	
(28)	CHAR Protected	40	SYSTEM_ LOG_REGISTER	Register of clients of RMSL.

--
-

A Client Register is just a chain of Identitys.

(28)	CHAR Protected	40	RMCR_CHAIN	
(28)	CHAR Private	4	*	
(30)	CHAR Protected	16	ITER0	
(30)	CHAR Private	4	*	
(38)	CHAR Protected	8	*	
(38)	ADDRESS Protected	4	PREV	
(3C)	ADDRESS Protected	4	NEXT	
(40)	CHAR Protected	16	NODE0	
(40)	CHAR Private	4	*	
(48)	CHAR Protected	8	*	
(48)	ADDRESS Protected	4	PREV	
(4C)	ADDRESS Protected	4	NEXT	
(50)	CHAR Protected	4	COLD_START_ CHAIN	System log chain token used for cold start.
(54)	UNSIGNED Protected	1	IN_COLD_STATE	Currently in cold start log records
(55)	CHAR Protected	40	*	

Constants

Len	Type	Value	Name	Description
4	CHARACTER	Init	RS_RESET	
4	CHARACTER	Cold	RS_COLD	
4	CHARACTER	DelP	RS_DELIVERY_ IN_PROGRESS	
4	CHARACTER	InKP	RS_KEYPOINT_ IN_PROGRESS	
4	CHARACTER	PreK	RS_PRE_KEYPOINT	
4	CHARACTER	Disj	RS_DISJOINT	
4	CHARACTER	KPDe	RS_KEYPOINT_ DELIVERY	
4	CHARACTER	Done	RS_COMPLETE	
4	CHARACTER		RMSL_NULL_CHAIN	
4	DECIMAL	1	RMSL_BUFFER_FULL	
4	DECIMAL	2	RMSL_INVALID_ DATA_LENGTH	

RMUW Recovery manager unit of work instance

-

The rmuw class is the Recovery Manager Unit of Work.

It may only be used by Recovery Manager. It is used to implement the RMUW gate.

rmuw inherits from rmlo and, via simulated inheritance, from rmpo and rmlg.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	1360	RMUW	
INSTANCE DATA				
Inherited Data				
(0)	CHAR Private	4	*	

-

The instance data of a RMUW object includes an instance of a Poller since the inheritance from Poller is simulated.

Offset Hex	Type	Len	Name (Dim)	Description
Declared Data				
(8)	CHAR Protected	1352	INSTANCE_ DATA_BLOCK	RMUW instance data
(8)	STRUCTURE IsA(RM_EYE_CATCHER) Protected	16	UOW_EYE_CATCHER	Eye-catcher
(8)	UNSIGNED Public	2	RM_EYE_LEN	object length
(A)	UNSIGNED Public	2	RM_EYE_OFFSET	offset of eye-catcher in object
(C)	CHAR Public	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(18)	OBJECT IsA(HOP_DCHAINNODE) Protected	16	UOW_CHAIN_LINK	Link in global UOW chain
(18)	CHAR Private	4	*	
(20)	CHAR Protected	8	*	
(20)	ADDRESS Protected	4	PREV	
(24)	ADDRESS Protected	4	NEXT	
(28)	STRUCTURE IsA(UOW_TOKEN_TYPE) Protected	4	UOW_TOKEN	UOW token
(2C)	UNSIGNED Protected	1	STATUS	UOW status
(2D)	FIXED Protected	1	LINKS_PRESENT	Whether links are left in the UOW
(2E)	UNSIGNED Protected	1	KEYPOINT_COUNT	# of keypoints seen
(2F)	UNSIGNED Protected	1	HEURISTIC_CAUSE	Cause of heurism
(30)	CHAR Protected	3	*	reserved

Offset Hex	Type	Len	Name (Dim)	Description
(33)	STRUCTURE IsA(RMUW_CONTEXT) Protected	31	UOW_CONTEXT	context info @POC
(33)	CHAR Protected	20	TRAN_CONTEXT	
(33)	CHAR Public	4	TERMID	Terminal id. of originating transaction
(37)	CHAR Public	8	TERMINAL_ LUNAME	Terminal LU name of originating transaction
(3F)	CHAR Public	4	TRANNUM	Transaction number of originating transaction
(43)	CHAR Public	4	TRANID	Transaction id. of originating transaction
(47)	CHAR Protected	8	*	
(47)	CHAR Protected	8	USERID	Userid of originating transaction
(47)	CHAR Protected	8	TRAN_TOKEN	Token for originating transaction
(4F)	CHAR Protected	3	OP_ID	Operator id. of originating transaction
(52)	UNSIGNED Protected	1	HEURISM	Whether to take a heuristic decision on an indoubt failure
(53)	UNSIGNED Protected	1	CHOICE	The default direction for a heuristic decision
(54)	UNSIGNED Protected	4	INDOUBT_ TIMEOUT_INTERVAL	Limit of amount of time and indoubt wait will be allowed befor being forced to take a heuristic decision. Zero denotes no time limit.
(58)	BITSTRING Protected	4	FLAGS	Flags.
(58)	BITSTRING Protected	1	*	
	1... .... Protected		FIRST_UOW_ FOR_TRANSACTION	First UOW for a transaction.
	.1.. .... Protected		RECONSTRUCTED	UOW was reconstructed during system restart.
	..1. .... Protected		SHUNTED	UOW is shunted.
	...1 .... Protected		HEURISTIC_ DECISION_TAKEN	A heuristic decision has been taken.
	.... 1... Protected		FORCE_PURGE_ PROTECTION	Protected from force purge.
	.... .1.. Protected		UNSHUNT_ ACTIVE	Unshunt in progress.
	.... ..1. Protected		RESYNCH_ IN_PROGRESS	Resynch. in progress.
	....1 Protected		EXISTENCE_ TO_BE_LOGGED	UOW existence needs logging.
(59)	BITSTRING Protected	1	*	
	1... .... Protected		EXISTENCE_ LOCKED	UOW may not be destroyed yet.
	.1.. .... Protected		RESUME_ REQUIRED	A transaction is suspended on this UOW.
	..1. .... Protected		UNSHUNT_ DEFERRED	Unshunt deferred until later.
	...1 .... Protected		SERIAL_ RECOVERY	UOW is being reconstructed during system restart but its indoubt or inflight log records have not yet been reached.
	.... 1... Protected		MOVE_IN_ PROGRESS	UOW is being moved on the log.
	.... .1.. Protected		LOCALLY_ COMMITTED	local commits done.
	.... ..1. Protected		KEYPOINTED_ FOR_MOVE	keypointed in order to move
	....1 Protected		LINKS_FORGOTTEN	no links left
(5A)	BITSTRING Protected	1	*	
	1... .... Protected		FIRST_COMMIT_ DONE	first attempt at commit completed
	.1.. .... Protected		TIMEOUT_ ACTIVE	Indoubt wait timeout is active for this UOW.
	..1. .... Protected		SURVIVED_ COLD_START	UOW has survived a cold start.
	...1 .... Protected		LOCAL_COMMIT_ LOGGED	logged the fact that UOW has locally comm- itted.
	.... 1... Protected		CLIENT_ STATE_RECOVERED	client state has been recovered
	.... .111 Protected		*	reserved
(5B)	BITSTRING Protected	1	*	
	1... .... Protected		USERID_FROZEN	userid cannot change
	.111 1111 Protected		*	reserved
(5C)	CHAR Protected	4	SYSTEM_ LOG_CHAIN_TOKEN	System log chain token for this UOW.
(60)	CHAR Protected	8	STATE_CHANGE_ TIME	Time of last change of state
(68)	OBJECT IsA(HOP_DCHAIN) Protected	40	UNSHUNT_Q	Queue of unshunt requests.
(68)	CHAR Private	4	*	

Offset Hex	Type	Len	Name (Dim)	Description
(70)	CHAR Protected	16	ITER0	
(70)	CHAR Private	4	*	
(78)	CHAR Protected	8	*	
(78)	ADDRESS Protected	4	PREV	
(7C)	ADDRESS Protected	4	NEXT	
(80)	CHAR Protected	16	NODE0	
(80)	CHAR Private	4	*	
(88)	CHAR Protected	8	*	
(88)	ADDRESS Protected	4	PREV	
(8C)	ADDRESS Protected	4	NEXT	
(90)	UNSIGNED Protected	4	SUSPEND_TOKEN	DS suspend token.
(94)	CHAR Protected	4	*	Reserved
(98)	OBJECT IsA(RMPO) Protected	32	POLLER	Poller instance.
(98)	CHAR Private	4	*	

--
-

vote is the result of the poll so far.

coordinator is the address of the coordinator voter or zero if there is no coordinator voter.

indoubt determines whether or not we are in the indoubt state. If we are indoubt, then there must be a coordinator voter otherwise there would be no way of resolving the indoubt.

resynchronisation_ in_progress records the resynchronisation state. This prevents multiple concurrent attempt to resynchronise and also protects us from a forced decision during resynchronisation.

read_ only is 'yes' if and only if all the voters polled so far have indicated that they are read-only.

continue is 'yes' if there will be a next UOW. Sometimes there will be a next UOW even when continue is 'no'. This is due to some voter preventing the next UOW from continuing even though the application requested it. In such cases, the next UOW is always aborted without the application having a chance to do further work.

(A0)	CHAR Protected	17	INSTANCE_ DATA_BLOCK	RMPO instance data
(A0)	ADDRESS Protected	4	COORDINATOR	coodinator voter for this poller
(A4)	UNSIGNED Protected	1	VOTE	result of polling so far
(A5)	UNSIGNED Protected	1	INDOUBT	whether or not poller is indoubt
(A6)	UNSIGNED Protected	1	RESYNCHRONISATION_ IN_PROGRESS	whether or not resynch. is in progress
(A7)	UNSIGNED Protected	1	READ_ONLY	read-only result of polling so far
(A8)	UNSIGNED Protected	1	CONTINUE	continuation result of polling so far
(A9)	CHAR Protected	8	*	
(B8)	OBJECT IsA(RMLS) Protected	112	LINKS	Set of links from this UOW to remote Recovery Managers.
(B8)	CHAR Private	4	*	

-

A Link Set object contains a chain of all the Links involved in this Unit of Work.

There are embedded Voter and Poller objects and a pointer to the Link picked as last-agent. A Link Set knows whether it is awaiting forget.

(C0)	CHAR Protected	98	INSTANCE_ DATA_BLOCK	
(C0)	CHAR Protected	40	RMLS_LINKS	Chain of link objects
(C0)	CHAR Private	4	*	
(C8)	CHAR Protected	16	ITER0	
(C8)	CHAR Private	4	*	
(D0)	CHAR Protected	8	*	

Offset Hex	Type	Len	Name (Dim)	Description
(D0)	ADDRESS Protected	4	PREV	
(D4)	ADDRESS Protected	4	NEXT	
(D8)	CHAR Protected	16	NODE0	
(D8)	CHAR Private	4	*	
(E0)	CHAR Protected	8	*	
(E0)	ADDRESS Protected	4	PREV	
(E4)	ADDRESS Protected	4	NEXT	
(E8)	ADDRESS Protected	4	RMLS_LAST_ LINK	Pointer to last agent or single updater link
(EC)	CHAR Protected	4	RMLS_VOTER	Voter Object
(EC)	CHAR Private	4	*	
(F0)	CHAR Protected	32	RMLS_POLLER	Poller Object
(F0)	CHAR Private	4	*	
(F8)	CHAR Protected	17	INSTANCE_DATA_BLOCK	RMPO instance data
(F8)	ADDRESS Protected	4	COORDINATOR	coodinator voter for this poller
(FC)	UNSIGNED Protected	1	VOTE	result of polling so far
(FD)	UNSIGNED Protected	1	INDOUBT	whether or not poller is indoubt
(FE)	UNSIGNED Protected	1	RESYNCHRONISATION_IN_PROGRESS	whether or not resynch. is in progress
(FF)	UNSIGNED Protected	1	READ_ONLY	read-only result of polling so far
(100)	UNSIGNED Protected	1	CONTINUE	continuation result of polling so far
(101)	CHAR Protected	8	*	
(110)	UNSIGNED Protected	1	RMLS_AWAITING_FORGET	Linkset is merely awaiting forget
(111)	BITSTRING Protected	1	RMLS_FLAGS	
	`1... .... Protected`		CHAIN_INITIALISED	Chain is initialised
	`.1.. .... Protected`		*	
	`..1. .... Protected`		LINK_COMMIT_ABENDED	A link abended during perform_commit
	`...1 .... Protected`		LINK_ROLLBACK_NOT_SUPPORTED	A rollback was tried on a link that does not support it.
(112)	CHAR Protected	8	RMLS_FAILURE_ TIME	Failure time
(11A)	CHAR Protected	8	*	
(128)	CHAR Protected	131	INLINE_ACCESS_STRUCTURE	Structure of values which may be accessed by inline macro expansions.
(128)	CHAR Protected	8	RMUX_LOCAL_ UOW_ID	
(130)	CHAR Protected	27	RMUX_REMOTE_UOW_ID	
(130)	UNSIGNED Protected	1	RMUX_REMOTE_ID_LENGTH	
(131)	UNSIGNED Protected	1	RMUX_REMOTE_ID_LU_NAME_LENGTH	
(132)	CHAR Protected	25	*	
(14B)	BITSTRING Protected	1	RMUX_FLAGS	
	`1... .... Protected`		OPTIMAL_CLIENTS_ONLY	
(14C)	ADDRESS Protected	4	RMUX_WORK_TOKEN_ARRAY (19)	
(198)	CHAR Protected	19	RMUX_CLIENT_ STATES	
(198)	BITSTRING Protected	1	CLIENT_STATE (19)	
	`1... .... Protected`		COMMIT_ COMPLETE	
	`.111 1111 Protected`		*	
(1AB)	CHAR Protected	4	*	reserved.
(1B0)	OBJECT IsA(RMRO) Protected	48	RO_ARRAY (19)	Resource Owner instances.
(1B0)	CHAR Private	4	*	
(1B8)	CHAR Protected	4	VOTER	
(1B8)	CHAR Private	4	*	

Offset Hex	Type	Len	Name (Dim)	Description

--
-

The RMRO instance is prepared by preparing the corresponding Resource Owner.

--
-

The RMRO instance is committed by committing the corresponding Resource Owner.

--
-

The instance data for a Resource Owner object includes its identity.

A type is declared for force tokens and a null force token is declared.

A log header type is declared the length field of which includes the length of the resource id. which is appended to the header structure. Whether or not there is a resource id. is indicated by the resource id. existence bit. The source field in the discriminant is always 'private' for a resource owner log record as this class is the source of the log record as far as the RM classes are concerned since RM doesn't own or understand the format of data which is passed on the APPEND function.

The backout structure is used during backout and backout retry to track the progress of backout. If the pointer to this structure is null, then either backout has not yet started or else backout has completed successfully. The backout structure itself is declared internally to the class as the users of the class should be insensitive to it.

The commit structure is used for forget processing. If the pointer to this structure is null, then there has been no request forget. The commit structure itself is declared internally to the class as the users of the class should be insensitive to it.

Offset Hex	Type	Len	Name (Dim)	Description
(1C0)	CHAR Protected	28	INSTANCE_ DATA_BLOCK	RMRO instance.
(1C0)	CHAR Protected	4	NAME	Resource Owner client name.
(1C4)	ADDRESS Protected	4	BACKOUT_ STRUCT	Pointer to backout failure structure.
(1C8)	ADDRESS Protected	4	COMMIT_STRUCT	Pointer to commit failure structure.
(1CC)	ADDRESS Protected	4	CLIENT_ IDENTITY_ADDRESS	Resource Owner client identity address.
(1D0)	BITSTRING Protected	1	SYSTEM_ RESTART_STATES	State during system restart.
	11.. .... Protected		COMMIT_STATE	Commit state.
	..11 1... Protected		BACKOUT_ STATE	Backout state.
	.... .11. Protected		REQ_FORGET_ STATE	Request forget state.
(1D1)	BITSTRING Protected	1	RO_CLIENT_ FLAGS	
	1... .... Protected		RECORDS_ IGNORED	Records ignored
	.111 1111 Protected		*	
(1D2)	CHAR Protected	10	*	
(540)	CHAR Protected	8	TIMER_TOKEN	TI domain indoubt wait timeout token
(548)	CHAR Protected	8	*	reserved for APAR fixes

SHARED DATA

Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE IsA(RM_TOKEN) Public	4	UOW_BROWSE_ TOKEN_TYPE	
(0)	CHAR Protected	57	UOW_BROWSE_ ELEMENT	
(0)	OBJECT IsA(HOP_DCHAINNODE) Protected	16	UOW_BROWSE_ CHAIN_LINK	

Inherited Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Private	4	*	
(8)	CHAR Protected	8	*	
(8)	ADDRESS Protected	4	PREV	
(C)	ADDRESS Protected	4	NEXT	

Offset Hex	Type	Len	Name (Dim)	Description
(10)	STRUCTURE IsA(UOW_BROWSE_TOKEN_TYPE) Protected	4	UOW_BROWSE_ TOKEN	
(18)	STRUCTURE IsA(ITERATOR) Protected	24	UOW_BROWSE_ITERATOR	
(18)	CHAR Public	16	ITERNODE	
(18)	CHAR Private	4	*	
(20)	CHAR Protected	8	*	
(20)	ADDRESS Protected	4	PREV	
(24)	ADDRESS Protected	4	NEXT	
(28)	ADDRESS Public	4	CURRNODE	
(2C)	ADDRESS Public	4	CHAIN_PTR	
(30)	CHAR Protected	4	UOW_BROWSE_ OWNER	
(34)	FIXED Protected	1	UOW_BROWSE_ ENDED	
(35)	CHAR Protected	2	UOW_BROWSE_ FILTER	
(35)	FIXED Protected	1	UOW_BROWSE_SHUNTED	
(36)	FIXED Protected	1	UOW_BROWSE_NOT_SHUNTED	
(37)	FIXED Protected	1	UOW_BROWSE_WORK_TOKEN	
(38)	UNSIGNED Protected	1	UOW_BROWSE_CLIENT_NAME	
(0)	FIXED Public	1	UNSHUNT_REASON	

The following request type contains a union of three different kinds of request: avail, indoubt resolution, and system restart requests. The union is empty except for avail requests.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Public	72	UNSHUNT_REQUEST	
(0)	OBJECT IsA(HOP_DCHAINNODE) Public	16	CHAIN_LINK	
(0)	CHAR Private	4	*	
(8)	CHAR Protected	8	*	
(8)	ADDRESS Protected	4	PREV	
(C)	ADDRESS Protected	4	NEXT	
(10)	FIXED Public	1	UREASON	
(11)	CHAR Public	3	*	
(14)	CHAR Public	52	*	
(14)	CHAR Public	52	AVAIL	
(14)	UNSIGNED Public	1	CLIENT_NAME	
(15)	FIXED Public	1	REMOVE	
(16)	CHAR VARY Public	45	LOCAL_ACCESS_ ID	
(45)	FIXED Public	1	GENERIC_LAI	
(46)	CHAR Public	2	*	

WARNING

The following declarations define the shape of parts of the RM log records. Careless changes would lead to the need to initial start CICS systems because the 'old' log records would be a different shape.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	17	RMUW_LOG_HEADER	
(0)	STRUCTURE IsA(RMLG_DISCRIMINANT) Protected	7	RMUW_LH_DISCRIMINANT	
(0)	UNSIGNED Public	2	RMLG_HEADER_LENGTH	
(2)	CHAR Public	1	RMLG_SOURCE	
(3)	CHAR Public	4	RMLG_NAME	
(7)	CHAR Protected	8	RMUW_LH_LOCAL_UOW_ID	
(F)	UNSIGNED Protected	1	RMUW_LH_ UOW_STATUS	
(10)	BITSTRING Protected	1	RMUW_LH_FLAGS	
	1... .... Protected		RMUW_LH_ HEURISM	
	.1.. .... Protected		RMUW_LH_CHOICE_FORWARD	
	..1. .... Protected		RMUW_LH_CONTEXT_PRESENT	
	...1 .... Protected		RMUW_LH_ CLIENT_STATE_PRESENT	
(11)	CHAR Protected		RMUW_LH_DATA	
(0)	CHAR Protected	9	RMUW_LOG_STATUS	
(0)	CHAR Protected	8	RMUW_LS_TIME	

Offset Hex	Type	Len	Name (Dim)	Description
(8)	UNSIGNED Protected	1	RMUW_LS_HEURISTIC_CAUSE	
(0)	CHAR Protected	31	RMUW_CONTEXT	
(0)	STRUCTURE IsA(RMXN_CONTEXT) Protected	20	TRAN_CONTEXT	
(0)	CHAR Public	4	TERMID	Terminal id. of originating transaction
(4)	CHAR Public	8	TERMINAL_ LUNAME	Terminal LU name of originating transaction
(C)	CHAR Public	4	TRANNUM	Transaction number of originating transaction
(10)	CHAR Public	4	TRANID	Transaction id. of originating transaction
(14)	CHAR Protected	8	*	
(14)	CHAR Protected	8	USERID	Userid of originating transaction
(14)	CHAR Protected	8	TRAN_TOKEN	Token for originating transaction
(1C)	CHAR Protected	3	OP_ID	Operator id. of originating transaction
(0)	CHAR Protected	67	RMUW_LOG_CONTEXT	
(0)	STRUCTURE IsA(RMUW_CONTEXT) Protected	31	RMUW_LC_UOW_CONTEXT	
(0)	CHAR Protected	20	TRAN_CONTEXT	
(0)	CHAR Public	4	TERMID	Terminal id. of originating transaction
(4)	CHAR Public	8	TERMINAL_ LUNAME	Terminal LU name of originating transaction
(C)	CHAR Public	4	TRANNUM	Transaction number of originating transaction
(10)	CHAR Public	4	TRANID	Transaction id. of originating transaction
(14)	CHAR Protected	8	*	
(14)	CHAR Protected	8	USERID	Userid of originating transaction
(14)	CHAR Protected	8	TRAN_TOKEN	Token for originating transaction
(1C)	CHAR Protected	3	OP_ID	Operator id. of originating transaction
(1F)	CHAR Protected	27	RMUW_LC_REMOTE_UOW_ID	
(3A)	CHAR Protected	8	RMUW_LC_TIME	
(42)	BITSTRING Protected	1	RMUW_LC_FLAGS	
	`1... .... Protected`		RMUW_LC_FIRST_UOW_FOR_TXN	
(0)	CHAR Protected	20	RMUW_LOG_CLIENT_STATE	
(0)	UNSIGNED Protected	1	RMUW_CS_COUNT	
(1)	CHAR Protected	19	RMUW_CS_STATES	

Constants

Len	Type	Value	Name	Description
4	CHARACTER		NULL_UOW_BROWSE_TOKEN	
1	DECIMAL	1	UNSHUNT_REASON_ AVAIL	
1	DECIMAL	2	UNSHUNT_REASON_INDOUBT_RES	
1	DECIMAL	3	UNSHUNT_REASON_RESTART	
4	CHARACTER		NULL_SYSTEM_LOG_CHAIN_TOKEN	
4	CHARACTER	STAT	STATUS_LOG_RECORD	
4	CHARACTER	EXIS	EXISTENCE_ LOG_RECORD	
4	CHARACTER	MOVE	KEYPOINT_MOVE_LOG_RECORD	
4	CHARACTER	COLD	LOCAL_COLD_LOG_RECORD	
4	DECIMAL	200	MNO_RECON_INDOUBT_UOWS	
4	DECIMAL	201	MNO_RECON_POST_COMMIT_UOWS	
4	DECIMAL	202	MNO_RECON_INFLIGHT_UOWS	
4	DECIMAL	203	MNO_SHUNTED_UOWS	
4	DECIMAL	204	MNO_NO_SHUNTED_ UOWS	
4	DECIMAL	205	MNO_SUCCESSFUL_KEYPOINT	
4	DECIMAL	228	MNO_RESYNC_INDOUBT_UOWS	
4	DECIMAL	229	MNO_RESYNC_CFAIL_BFAIL_UOWS	
4	DECIMAL	230	MNO_RESYNC_INFLIGHT_UOWS	
4	DECIMAL	400	MNO_INCOMPLETE_UOW_ERROR	
8	CHARACTER	RM0400	DCD_INCOMPLETE_UOW_ERROR	
4	DECIMAL	1	RMUW_BUFFER_FULL	
4	DECIMAL	2	RMUW_INVALID_DATA_LENGTH	
4	CHARACTER		NULL_UOW_TOKEN	
4	CHARACTER	RMUW	UOW_LOGGABLE_ ID_NAME	

Len	Type	Value	Name	Description
4	DECIMAL	301	MNO_FORCE_ PURGE_REJECTED	

RMUW Recovery manager unit of work class data

-

This is the declaration for the rmuw_class_data class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	2520	RMUW_CLASS_DATA	

-

The UOW class data consists of some types, the address of a pro-forma UOW, a couple of token sets for UOW and UOW browse tokens, respectively, a chain of UOWs, a chain of UOW browses, a UOW factory, and a register of UOW log clients.

INSTANCE DATA

Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	2516	CLASS_DATA_BLOCK	RMUW class data
(0)	STRUCTURE IsA(RM_EYE_CATCHER) Protected	16	UOW_CD_ EYE_CATCHER	
				Eye-catcher
(0)	UNSIGNED Public	2	RM_EYE_LEN	object length
(2)	UNSIGNED Public	2	RM_EYE_OFFSET	offset of eye-catcher in object
(4)	CHAR Public	12	RM_EYE_STRING	'>DFHRMxxxxxx'
(10)	ADDRESS Protected	4	PROFORMA_ UOW_POINTER	
				Pro-forma UOW address
(14)	CHAR Protected	4	*	Reserved
(18)	OBJECT IsA(HOP_DCHAIN) Protected	40	UOW_CHAIN	Global UOW chain

Inherited Data

Offset Hex	Type	Len	Name (Dim)	Description
(18)	CHAR Private	4	*	
(20)	CHAR Protected	16	ITER0	
(20)	CHAR Private	4	*	
(28)	CHAR Protected	8	*	
(28)	ADDRESS Protected	4	PREV	
(2C)	ADDRESS Protected	4	NEXT	
(30)	CHAR Protected	16	NODE0	
(30)	CHAR Private	4	*	
(38)	CHAR Protected	8	*	
(38)	ADDRESS Protected	4	PREV	
(3C)	ADDRESS Protected	4	NEXT	
(40)	OBJECT IsA(RMOF) Protected	40	UOW_FACTORY	UOW factory

-

The instance data contains an eye-catcher, a subpool name, and a subpool token. The subpool name is used as a remark when allocating and freeing storage. It consists of the prefix 'RMOF' and a suffix which is the name of the object being managed.

Offset Hex	Type	Len	Name (Dim)	Description
(40)	CHAR Protected	40	INSTANCE_ DATA_BLOCK	
				RMOF instance data
(40)	CHAR Protected	16	OF_EYE_ CATCHER	eye-catcher
(40)	UNSIGNED Public	2	RM_EYE_LEN	object length
(42)	UNSIGNED Public	2	RM_EYE_ OFFSET	offset of eye-catcher in object
(44)	CHAR Public	12	RM_EYE_ STRING	'>DFHRMxxxxxx'
(50)	CHAR Protected	8	SUBPOOL_NAME	subpool name

Offset Hex	Type	Len	Name (Dim)	Description
(50)	CHAR Protected	4	SUBPOOL_ NAME_PREFIX	subpool name prefix
(54)	CHAR Protected	4	SUBPOOL_ NAME_SUFFIX	subpool name suffix
(58)	CHAR Protected	8	SUBPOOL_TOKEN	subpool token
(60)	CHAR Protected	8	*	
(68)	OBJECT IsA(RMLI) Protected	88	UOW_LOGGABLE_ID	Loggable id. of RMUW with respect to RMSL
(68)	CHAR Private	4	*	
(70)	CHAR Protected	8	*	
(70)	ADDRESS Protected	4	PREV	
(74)	ADDRESS Protected	4	NEXT	

--
-

The only piece of instance data is the name of the identity.

(78)	CHAR Protected	4	NAME	

--
-

The instance data, in addition to that inherited from the rmid class, consists of the address of the start delivery, deliver data, end delivery, take keypoint, set chain token, and inquire disjoint chains methods of an instance of (a subclass of) the loggable object class.

(80)	CHAR Protected	64	INSTANCE_ DATA_BLOCK	RMLI instance data.
(80)	ADDRESS Protected	4	START_DELIVERY	Start delivery method address.
(84)	ADDRESS Protected	4	DELIVER_DATA	Deliver data method address.
(88)	ADDRESS Protected	4	END_DELIVERY	End delivery method address.
(8C)	ADDRESS Protected	4	TAKE_KEYPOINT	Take keypoint method address.
(90)	ADDRESS Protected	4	SET_CHAIN_ TOKEN	Set chain token method address.
(94)	ADDRESS Protected	4	INQUIRE_ DISJOINT_CHAINS	Inquire disjoint chains method address.
(98)	ADDRESS Protected	4	PRE_KEYPOINT	Start Keypoint method address.
(9C)	ADDRESS Protected	4	POST_KEYPOINT	Start Keypoint method address.
(A0)	CHAR Protected	32	*	
(C0)	OBJECT IsA(RMCR) Protected	40	UOW_LOG_ REGISTER	Register of clients of the RMUW log

--
-

A Client Register is just a chain of Identitys.

(C0)	CHAR Protected	40	RMCR_CHAIN	
(C0)	CHAR Private	4	*	
(C8)	CHAR Protected	16	ITER0	
(C8)	CHAR Private	4	*	
(D0)	CHAR Protected	8	*	
(D0)	ADDRESS Protected	4	PREV	
(D4)	ADDRESS Protected	4	NEXT	
(D8)	CHAR Protected	16	NODE0	
(D8)	CHAR Private	4	*	
(E0)	CHAR Protected	8	*	
(E0)	ADDRESS Protected	4	PREV	
(E4)	ADDRESS Protected	4	NEXT	
(E8)	CHAR Protected	19	UOW_RO_ SYNCPOINT_ ORDER_ARRAY	
(E8)	UNSIGNED Protected	1	UOW_RO_ SYNCPOINT_ORDER (19)	Array defining the order in which RO clients are called in syncpoint

Offset Hex	Type	Len	Name (Dim)	Description
(100)	OBJECT IsA(RMTOKSET) Protected	1056	UOW_TOKEN_SET	Set of UOW tokens

--
-

The token set records the set of known tokens together with the address associated with each known token.

Offset Hex	Type	Len	Name (Dim)	Description
(100)	CHAR Protected	1056	INSTANCE_ DATA_BLOCK	
(100)	CHAR Protected	16	EYE_CATCHER	eyecatcher
(100)	UNSIGNED Public	2	RM_EYE_LEN	object length
(102)	UNSIGNED Public	2	RM_EYE_ OFFSET	offset of eye-catcher in object
(104)	CHAR Public	12	RM_EYE_ STRING	'>DFHRMxxxxxx'
(110)	UNSIGNED Protected	4	NUMBER_ OF_BLOCKS	block count
(114)	UNSIGNED Protected	4	FREE_CHAIN_ HEAD	free chain head
(114)	CHAR Protected	2	INDEX	
(114)	UNSIGNED Protected	1	BLOCK	
(115)	UNSIGNED Protected	1	SLOT	
(116)	UNSIGNED Protected	2	INSTANCE	
(118)	ADDRESS Protected	4	BLOCKS (0 255)	pointers to blocks
(518)	CHAR Protected	8	*	
(520)	OBJECT IsA(RMTOKSET) Protected	1056	UOW_BROWSE_ TOKEN_SET	Set of UOW browse tokens
(520)	CHAR Protected	1056	INSTANCE_ DATA_BLOCK	
(520)	CHAR Protected	16	EYE_CATCHER	eyecatcher
(520)	UNSIGNED Public	2	RM_EYE_LEN	object length
(522)	UNSIGNED Public	2	RM_EYE_ OFFSET	offset of eye-catcher in object
(524)	CHAR Public	12	RM_EYE_ STRING	'>DFHRMxxxxxx'
(530)	UNSIGNED Protected	4	NUMBER_ OF_BLOCKS	block count
(534)	UNSIGNED Protected	4	FREE_CHAIN_ HEAD	free chain head
(534)	CHAR Protected	2	INDEX	
(534)	UNSIGNED Protected	1	BLOCK	
(535)	UNSIGNED Protected	1	SLOT	
(536)	UNSIGNED Protected	2	INSTANCE	
(538)	ADDRESS Protected	4	BLOCKS (0 255)	pointers to blocks
(938)	CHAR Protected	8	*	
(940)	OBJECT IsA(HOP_DCHAIN) Protected	40	UOW_BROWSES	Chain of UOW browses.
(940)	CHAR Private	4	*	
(948)	CHAR Protected	16	ITER0	
(948)	CHAR Private	4	*	
(950)	CHAR Protected	8	*	
(950)	ADDRESS Protected	4	PREV	
(954)	ADDRESS Protected	4	NEXT	
(958)	CHAR Protected	16	NODE0	
(958)	CHAR Private	4	*	
(960)	CHAR Protected	8	*	
(960)	ADDRESS Protected	4	PREV	
(964)	ADDRESS Protected	4	NEXT	
(968)	CHAR Protected	68	UOW_STATISTICS	UOW-related statistics:
(968)	SIGNED Protected	4	TOTAL_SYNC_ FWDS	#forward commits
(96C)	SIGNED Protected	4	TOTAL_SYNC_ BWDS	#backward commits
(970)	CHAR Protected	8	TOTAL_TIME_ SHUNTED_INDOUBT	

Offset Hex	Type	Len	Name (Dim)	Description
	total time UOWs were shunted indoubt			
(978)	SIGNED Protected	4	TOTAL_SHUNTED_INDOUBT	
				#unshunts of indoubt UOWs
(97C)	SIGNED Protected	4	TOTAL_SHUNTED_RO_FAIL	
				#final unshunts of
	backout or commit failed UOWs			
(980)	CHAR Protected	8	TOTAL_TIME_SHUNTED_RO_FAIL	
	total time backout or commit failed UOWs were shunted			
	The following fields count the number of heuristic decisions due to particular reasons.			
(988)	SIGNED Protected	4	HEURISM_FORCED_BY_TRANDEF	
				#due to txn defn
(98C)	SIGNED Protected	4	HEURISM_FORCED_BY_TIMEOUT	
				#due to timeout
(990)	SIGNED Protected	4	HEURISM_ FORCED_BY_OPERATOR	
				#due to operator
(994)	SIGNED Protected	4	HEURISM_FORCED_BY_OTHER	
				#due to other reason
	The following fields count the number of heuristic decisions forced by a client of the UOW			
(998)	SIGNED Protected	4	HEURISM_ FORCED_BY_CLIENT_TD	
				#due to TD
(99C)	SIGNED Protected	4	HEURISM_ FORCED_BY_CLIENT_LU61	
				#due to LU 6.1
(9A0)	SIGNED Protected	4	HEURISM_ FORCED_BY_CLIENT_MRO	
				#due to MRO
(9A4)	SIGNED Protected	4	HEURISM_ FORCED_BY_CLIENT_RMI	
				#due to RMI
(9A8)	SIGNED Protected	4	HEURISM_ FORCED_BY_CLIENT_OTHER	
				#due to other client
(9AC)	CHAR Protected	40	*	reserved for APAR fixes
SHARED DATA				
Declared Data				
(0)	CHAR Public	4	UOW_TOKEN_TYPE	

Constants

Len	Type	Value	Name	Description
4	CHARACTER		NULL_UOW_TOKEN	
4	CHARACTER	RMUW	UOW_LOGGABLE_ ID_NAME	
4	DECIMAL	301	MNO_FORCE_PURGE_REJECTED	
4	CHARACTER		NULL_UOW_BROWSE_TOKEN	
1	DECIMAL	1	UNSHUNT_REASON_ AVAIL	
1	DECIMAL	2	UNSHUNT_REASON_INDOUBT_RES	
1	DECIMAL	3	UNSHUNT_REASON_RESTART	
4	CHARACTER		NULL_SYSTEM_LOG_CHAIN_TOKEN	
4	CHARACTER	STAT	STATUS_LOG_RECORD	
4	CHARACTER	EXIS	EXISTENCE_ LOG_RECORD	
4	CHARACTER	MOVE	KEYPOINT_MOVE_LOG_RECORD	
4	CHARACTER	COLD	LOCAL_COLD_LOG_RECORD	
4	DECIMAL	200	MNO_RECON_INDOUBT_UOWS	
4	DECIMAL	201	MNO_RECON_POST_COMMIT_UOWS	
4	DECIMAL	202	MNO_RECON_INFLIGHT_UOWS	
4	DECIMAL	203	MNO_SHUNTED_UOWS	
4	DECIMAL	204	MNO_NO_SHUNTED_ UOWS	

Len	Type	Value	Name	Description
4	DECIMAL	205	MNO_SUCCESSFUL_ KEYPOINT	
4	DECIMAL	228	MNO_RESYNC_ INDOUBT_UOWS	
4	DECIMAL	229	MNO_RESYNC_ CFAIL_BFAIL_UOWS	
4	DECIMAL	230	MNO_RESYNC_ INFLIGHT_UOWS	
4	DECIMAL	400	MNO_INCOMPLETE_ UOW_ERROR	
8	CHARACTER	RM0400	DCD_INCOMPLETE_ UOW_ERROR	
4	DECIMAL	1	RMUW_BUFFER_FULL	
4	DECIMAL	2	RMUW_INVALID_ DATA_LENGTH	

RRAB Resource definition recovery definitions

CONTROL BLOCK NAME = DFHRRAB
DESCRIPTIVE NAME = CICS Resource definition Recovery Anchor
FUNCTION =
    DFHRRAB describes the DSECT for the Resource definition
    Recovery Anchor Block. This block serves as an anchor
    for the set of Resource Recovery Anchor Blocks with Names
    (RABNs) and also two action-lists containing Resource
    Definition Action Lists (RDALs). These action-lists
    and RABNs describe the work undertaken during an Install
    process for communication resources (terminals,
    typeterms, connections and sessions). It also contains a
    flag which indicates whether Terminal Object Resolution
    needs to be driven at the end of the UOW. There is only
    one RRAB for each UOW, fresh requests reuse an existing
    RRAB.
    The RRAB also points to a list of Resource definition
    update blocks which list the definitions that have been
    locked during this UOW. This list is checked before an
    add to ensure that we are not attempting to add a defn
    which another UOW is attempting to delete.
    The Resource definition Recovery Anchor Block is built by
    Table Builder Services as part of the processing of an
    Install (or Delete) request. It is also built by Terminal
    Object Resolution during Install or Delete Requests. It is
    used as the Recovery Manager Client token for 'APRD'.
    The Resource definition Recovery Anchor Block is deleted
    when all the action-lists and RABN chain are empty the TOR
    flag is reset, the RDUB chain is empty, eith by TBS, TOR or
    DFHAPRD. At the same time Recovery Manager token is reset
    to zero.
LIFETIME =
    Created when the first Table Builder or Terminal Object
    Resolution request that is recoverable is processed, or a
    lock is obtained.
    Deleted at end of transaction.
STORAGE CLASS =
    Above 16M line.
LOCATION =
    Issuing an INQUIRE_WORK_TOKEN to the recovery manager
    with Client Name 'APRD' returns the address of the
    Resource Recovery Anchor Block.
-

This is the definition of the RRAB

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	41	DFHRRAB	
(0)	CHARACTER	8	RRAB_HDR	set to >DFHRRAB
(8)	ADDRESS	4	RRAB_CURRENT_ ACTION_LIST	ptr to non-atom current actions
(C)	ADDRESS	4	RRAB_CURRENT_ ACTION_LIST_END	ptr to end non-atom current actions
(10)	ADDRESS	4	RRAB_NAMED_LIST	ptr to rabn chain
(14)	ADDRESS	4	RRAB_CURRENT_ RABN	ptr to current rabn

Offset Hex	Type	Len	Name (Dim)	Description
(18)	ADDRESS	4	RRAB_DELAYED_ ACTION_LIST	ptr to non-atom actions for sync
(1C)	ADDRESS	4	RRAB_DELAYED_ ACTION_LIST_END	ptr to end non-atom actions for sync
(20)	ADDRESS	4	RRAB_RDUB	ptr to RDUBs
(24)	ADDRESS	4	RRAB_LAST_RDUB	ptr to RDUBs end
(28)	BITSTRING	1	RRAB_BITS	RAB flags
	1...		RRAB_TOR	1 means TOR interest
	.1..		RRAB_OPEN	1 means RAB active for TBS
	..1.		RRAB_FORGET	1 means RAB active for restart
	...1 1111		*	Reserved

--
CONTROL BLOCK NAME = DFHRABN
DESCRIPTIVE NAME = CICS Resource Recovery Atom Block Name
SOURCE = DFHRRAB DESIGN part of DFHAPRDR DESIGN
FUNCTION =

DFHRABN describes the DSECT for the Resource Recovery Atom Block Name. This block serves as an anchor for an action-list. It defines the set of actions that are performed for a named 'atom' of resource recovery for either a Pipe-Line or a Connection definition. It retains a flag that describes the back-out of the atom in case further actions for that atom arrive, so that they can be prevented.

The Resource Recovery Atom Block Name is built by Table Builder Services as part of the processing of an Install request. It is added to a chain from the Resource definition Recovery Anchor Block (RRAB), and pointed to as the active RABN.

The Resource Recovery Anchor Block is deleted when an END_ATOMS call is made or the UOW ends. The action-list is transfered to the delayed-action-list on the RRAB.

LIFETIME =

Created when the first Table Builder or Terminal Object Resolution request that is recoverable for an atom is processed.
Deleted at end of a UOW.

STORAGE CLASS =
Above 16M line.
LOCATION =
Chained from the RRAB.
INNER CONTROL BLOCKS =
None.
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Control block definition
EXTERNAL REFERENCES = None
DATA AREAS = None
CONTROL BLOCKS = None
GLOBAL VARIABLES (Macro pass) = None

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	DFHRABN	
(0)	CHARACTER	8	RABN_HEADER	Set to >DFHRABN
(8)	ADDRESS	4	RABN_FWD_PTR	RABN chain ptr
(C)	CHARACTER	9	RABN_ATOM_ID	Name of atom
(15)	BITSTRING	1	RABN_BITS	Flag bit for RABN
	1...		RABN_BACKED_OUT	1 means backout atom
	.111 1111		*	Reserved
(16)	BITSTRING	2	*	Reserved
(18)	ADDRESS	4	RABN_ACTION_LIST	ptr to action list
(1C)	ADDRESS	4	RABN_ACTION_ LIST_END	ptr to end action

Constants

Len	Type	Value	Name	Description
8	CHARACTER	>DFHRRAB	RRAB_NAME	
8	CHARACTER	>DFHRABN	RABN_NAME	

RUEI Logger reusable extended iliffe vector class

-

The RUEI and MRUEI classes are both collected into the DFHLGUDC copybook which may then in turn be included by calling code.

-

RUEI is the Reusable Extended Iliffe Vector class.

Before declaring this class, the user should declare a constant RUEI_SIZE to indicate the number of elements which may be set in this particular vector.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	20	RUEI	

--
-

The reusable extended Iliffe vector contains an array of elements and a continuation pointer. Note that the continuation pointer follows the last element so that the browse need not record the current vector address as well as the current element address.

The vector also contains two sums of element lengths. One sum (ruei_ elem_length_ sum) contains the total length of data elements pointed to by this vector alone. The other sum (ruei_ elem_length_ sum_sum) contains the sum of lengths of data elements in this vector plus the lengths of all the elements pointed to in the linked list of vectors pointed to by this ruei.

Finally, a public constant is included to denote the end of a browse.

INSTANCE DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	20	*	
(0)	UNSIGNED Protected	4	RUEI_ELEM_ LENGTH_SUM	
(4)	UNSIGNED Protected	4	RUEI_ELEM_ LENGTH_SUM_SUM	
(8)	CHAR Protected	8	RUEI_ELEMS (1)	
(8)	ADDRESS Protected	4	RUEI_ELEM_ADDR	
(8)	BITSTRING Protected	1	*	
	`1... .... Protected`		RUEI_ELEM_ ADDR_FLAG	OFF means this is NOT a continuation pointer
(C)	UNSIGNED Protected	4	RUEI_ELEM_ LENGTH	
(10)	ADDRESS Protected	4	RUEI_CONTINUATION	Zero pointer means there there is no continuation to this vector. Non-zero values point to the continuation of this vector.
(10)	BITSTRING Protected	1	*	
	`1... .... Protected`		RUEI_CONTINUATION_ FLAG	ON means this is a continuation pointer

Constants

Len	Type	Value	Name	Description
4	DECIMAL	2147483647	RUEI_BROWSE_END	

SHRTC Sh request routing class

Offset Hex	Type	Len	Name (Dim)	Description
1	CHARACTER	0	ROUTE_SELECT	
1	CHARACTER	1	ROUTE_ERROR	
1	CHARACTER	2	ROUTE_TERMINATE	
1	CHARACTER	3	ROUTE_NOTIFY	
1	CHARACTER	4	ROUTE_ABEND	
1	CHARACTER	5	ROUTE_INITIATE	
1	CHARACTER	6	ROUTE_COMPLETE	
1	CHARACTER	0	SH_SYSID_NOT_FOUND	
1	CHARACTER	1	SH_SYSID_ OUT_SERVICE	
1	CHARACTER	2	SH_NO_SESSIONS	
1	CHARACTER	3	SH_ALLOCATE_ REJECTED	
1	CHARACTER	4	SH_QUEUE_PURGED	
1	CHARACTER	5	SH_FUNC_NOT_ SUPPORTED	
1	CHARACTER	6	SH_LENGERR	
1	CHARACTER	7	SH_PGMIDERR	
1	CHARACTER	8	SH_INVREQ	
1	CHARACTER	9	SH_NOTAUTH	
1	CHARACTER	A	SH_TERMERR	
1	CHARACTER	B	SH_ROLLEDBACK	
1	CHARACTER	C	SH_TRANSIDERR	
1	CHARACTER	D	SH_IOERR	
1	CHARACTER	E	SH_USERIDERR	
1	CHARACTER	0	TRADITIONAL_ ROUTING	
1	CHARACTER	1	NOTIFY_REQUEST	
1	CHARACTER	2	START_NO_ DATA_REQUEST	
1	CHARACTER	3	START_WITH_ DATA_REQUEST	
1	CHARACTER	4	DPL_REQUEST	
1	CHARACTER	5	CBTS_REQUEST	
1	CHARACTER	6	NON_TERM_ START_REQUEST	

--
--

12	CHARACTER	>DFHSHRDATA	RDATA_EYECATCHER_ STRING	
8	CHARACTER		UCMASK	

SMDCC Storage manager anchor block

SMA - SM Anchor block
This block contains the global storage for the SM domain.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	476	SMA	
(0)	CHARACTER	16	SMA_PREFIX	
(0)	HALFWORD	2	SMA_LENGTH	
(2)	CHARACTER	1	SMA_ARROW	
(3)	CHARACTER	3	SMA_DFH	
(6)	CHARACTER	2	SMA_DOMID	
(8)	CHARACTER	8	SMA_BLOCK_NAME	
(10)	ADDRESS	4	SMA_SCQFREEHEAD	-> first free SCQ
(14)	ADDRESS	4	SMA_SCAFREEHEAD	-> first free SCA
(18)	CHARACTER	8	*	header for task SCA chain
(18)	ADDRESS	4	SMA_SCA_ TASK_FIRST	-> first task SCA
(1C)	ADDRESS	4	SMA_SCA_ TASK_LAST	-> last task SCA
(20)	CHARACTER	8	*	header for domain SCA chain
(20)	ADDRESS	4	SMA_SCA_DOMAIN_FIRST	-> first domain SCA
(24)	ADDRESS	4	SMA_SCA_ DOMAIN_LAST	-> last domain SCA
(28)	FULLWORD	4	SMA_SCANUM	current SCA number
(2C)	FULLWORD	4	SMA_SPIDNUM	current spid number
(30)	ADDRESS	4	SMA_SMXFREEHEAD	SMX freechain
(34)	CHARACTER	8	*	allocated SMX chain
(34)	ADDRESS	4	SMA_SMX_FIRST	-> first allocated SMX
(38)	ADDRESS	4	SMA_SMX_LAST	-> last allocated SMX
(3C)	ADDRESS	4	SMA_SMLOCK	SM lock token
(40)	BITSTRING	1	SMA_FLAGS	flags
	1...		SMA_SMSY_ RESUMED	='1'B, system task resumed
	.1..		SMA_STORAGE_RECOVERY	
	..1.		SMA_STORAGE_PROTECT_REQ	
	...1		SMA_STORAGE_PROTECT	
	.... 1...		SMA_REENTRANT_PROGRAM_PROTECT	
	.... .1..		SMA_TRANSACTION_ISOLATION_REQ	
	.... ..1.		SMA_LOC_ EXPLICIT	
	1		SMA_NOTIFIED_DSAS_NOT_CONSTRAINED	
(41)	UNSIGNED	1	SMA_SM_STATE	SM domain state
(42)	BITSTRING	1	SMA_FLAGS2	
	1...		SMA_SOS_BELOW	='1'b, SOS below 16MB
	.1..		SMA_SOS_ABOVE	='1'b, SOS above 16MB
	..11 1111		*	reserved
(43)	BITSTRING	1	SMA_DSAS_FIXED	fixed DSAs
	1...		SMA_CDSA_FIXED	CDSA fixed
	.1..		SMA_UDSA_FIXED	UDSA fixed
	..1.		SMA_SDSA_FIXED	SDSA fixed
	...1		SMA_RDSA_FIXED	RDSA fixed
	.... 1...		SMA_ECDSA_FIXED	ECDSA fixed
	.... .1..		SMA_EUDSA_FIXED	EUDSA fixed
	.... ..1.		SMA_ESDSA_FIXED	ESDSA fixed
	1		SMA_ERDSA_FIXED	ERDSA fixed
(44)	ADDRESS	4	SMA_SCABLOCKHEAD	head of SCA block chain
(48)	ADDRESS	4	SMA_SCQBLOCKHEAD	head of SCQ block chain
(4C)	ADDRESS	4	SMA_SMXBLOCKHEAD	head of SMX block chain
(50)	ADDRESS	4	SMA_MCAP	-> macro-compat anchor
(54)	ADDRESS	4	SMA_SQEBLOCKHEAD	-> SQE block head
(58)	ADDRESS	4	SMA_SQEFREEHEAD	-> SQE free chain head
(5C)	FULLWORD	4	SMA_SYSTEM_TASK_RUNS	
(60)	FULLWORD	4	SMA_SYSTEM_TASK_NOTIFIES	
(64)	ADDRESS	4	SMA_SYSTEM_SUSPEND_TOKEN	
(68)	CHARACTER	8	SMA_LAST_ RESET_TIME	time of last Stats reset
(70)	FULLWORD	4	*	Reserved
(74)	FULLWORD	4	SMA_SQE_COUNT	number of SQEs
(78)	FULLWORD	4	SMA_SMX_COUNT	number of SMXs
(7C)	CHARACTER	8	*	
(7C)	ADDRESS	4	SMA_PPA_FIRST	-> first PPA

Offset Hex	Type	Len	Name (Dim)	Description
(80)	ADDRESS	4	SMA_PPA_LAST	-> last PPA
(84)	ADDRESS	4	SMA_PPA_ BELOW_HEAD	-> first below 16MB PPA
(88)	ADDRESS	4	SMA_PPA_ ABOVE_HEAD	-> first above 16MB PPA
Following array holds values for each of the DSAs.				
(8C)	CHARACTER	16	* (8)	
(8C)	ADDRESS	4	SMA_PPAP	-> PPA
(90)	FULLWORD	4	SMA_PRIMARY_EXTENT_SIZE	primary extent size
(94)	FULLWORD	4	*	reserved
(98)	FULLWORD	4	*	reserved
(10C)	FULLWORD	4	SMA_SUSPENDED	total suspended reqsts
(110)	ADDRESS	4	SMA_SATP	-> storage access table
(114)	ADDRESS	4	SMA_STATS_ BUFFER_PTR	Stats buffer address
(118)	FULLWORD	4	SMA_DSA_LIMIT	DSALIMIT value
(11C)	FULLWORD	4	SMA_EDSA_LIMIT	EDSALIMIT value
(120)	CHARACTER	8	SMA_SQEHEAD	
(120)	ADDRESS	4	SMA_SQE_FIRST	-> first SQE
(124)	ADDRESS	4	SMA_SQE_LAST	-> last SQE
(128)	ADDRESS	4	SMA_DXHP	-> DXH
(12C)	UNSIGNED	4	SMA_DSA_ CURRENT_SIZE	current total DSA storage
(130)	UNSIGNED	4	SMA_EDSA_CURRENT_SIZE	current total EDSA storge
(134)	ADDRESS	4	SMA_CTNFREEHEAD	-> first free CTN
(138)	FULLWORD	4	SMA_DSA_ NON_EMPTY	non-empty DSA extent stg
(13C)	FULLWORD	4	SMA_EDSA_ NON_EMPTY	non-empty EDSA extent stg
(140)	FULLWORD	4	*	reserved
Subspace Manager related fields.				
(144)	ADDRESS	4	SMA_SUABLOCKHEAD	-> SUA blocks
(148)	ADDRESS	4	SMA_SUA_FREEHEAD	-> SUA free chain
(14C)	CHARACTER	8	* (0 1)	Array of SUA pool chains
(14C)	ADDRESS	4	SMA_SUA_ POOL_FIRST	-> first SUA
(150)	ADDRESS	4	SMA_SUA_ POOL_LAST	-> last SUA
(15C)	CHARACTER	8	*	SUA allocated chain
(15C)	ADDRESS	4	SMA_SUA_ ALLOC_FIRST	-> first SUA
(160)	ADDRESS	4	SMA_SUA_ ALLOC_LAST	-> last SUA
(164)	CHARACTER	8	*	SUA steal chain
(164)	ADDRESS	4	SMA_SUA_ STEAL_FIRST	-> first SUA
(168)	ADDRESS	4	SMA_SUA_ STEAL_LAST	-> last SUA
(16C)	ADDRESS	4	SMA_COMMON_SUA_ADDRESS	-> common SUA
(170)	UNSIGNED	2	SMA_SUA_ FREE_COUNT	SUA free count
(172)	UNSIGNED	2	SMA_SUA_ALL_POOLS_COUNT	SUA count for all pools
(174)	CHARACTER	4	* (0 1)	
(174)	UNSIGNED	2	SMA_SUA_ POOL_COUNT	SUA pool count
(176)	UNSIGNED	2	SMA_SUA_ POOL_MIN	LWM of pool for interval
(17C)	UNSIGNED	2	SMA_SUA_ALLOCATED_COUNT	SUA allocated count
(17E)	UNSIGNED	2	SMA_SUA_POOL_AVG	Weighted average of the no. of SUAs on pool chains
(180)	UNSIGNED	4	SMA_ALET_LIMIT	Maximum number of ALETs
(184)	UNSIGNED	4	SMA_ALET_COUNT	Number of ALETs in use
Do not alter the structure below without altering DFHSMSRI.				
(188)	CHARACTER	8	SMA_ISOLATION_ STRUC	Isolation token structure
(188)	BITSTRING	1	SMA_ISOLATION_ FLAGS	
	1...		SMA_TRANSACTION_ISOLATION	='1' TRANISO active
	.111 1111		*	Reserved
(189)	CHARACTER	3	*	Reserved
(18C)	ADDRESS	4	SMA_QR_TCB	QR TCB ptr
(190)	CHARACTER	40	*	Statistics related fields
(190)	FULLWORD	4	SMA_COMMON_SS_CUMULATIVE_ USERS	Cummmulative number of common subspace users.
(194)	FULLWORD	4	SMA_COMMON_SS_CURRENT_USERS	Current number of common subspace users.
(198)	FULLWORD	4	SMA_COMMON_SS_HWM_OF_USERS	

Offset Hex	Type	Len	Name (Dim)	Description
				High water mark of common subspace users
(19C)	FULLWORD	4	SMA_UNIQUE_ SS_CUMULATIVE_ USERS	Cummulative number of unique subspace users.
(1A0)	FULLWORD	4	SMA_UNIQUE_ SS_CURRENT_USERS	Current number of unique subspace users.
(1A4)	FULLWORD	4	SMA_UNIQUE_ SS_HWM_OF_USERS	High water mark of unique subspace users.
(1A8)	FULLWORD	4	SMA_CUMULATIVE_ ALET_STEALS	Cummulative number of ALETs stolen.
(1AC)	FULLWORD	4	SMA_ACTIVE_ TASK_ALET_STEALS	Number of ALETs stolen from active tasks.
(1B0)	FULLWORD	4	SMA_NUMBER_ OF_SS_CREATES	Number of IARSUBSP create calls.
(1B4)	FULLWORD	4	SMA_NUMBER_ OF_SS_DELETES	Number of IARSUBSP delete calls.
(1B8)	UNSIGNED	4	SMA_DSA_ LIMIT_STORAGE	actual DSALIMIT storage
(1BC)	UNSIGNED	4	SMA_EDSA_ LIMIT_STORAGE	actual EDSALIMIT storage
(1C0)	UNSIGNED	4	SMA_HWM_DSA_SIZE	hwm total dsa storage
(1C4)	UNSIGNED	4	SMA_HWM_ EDSA_SIZE	hwm total edsa storage
(1C8)	CHARACTER	8	SMA_LAST_ TUNING_TIME	time self-tuning subpool stats were last updated
(1D0)	FULLWORD	4	*	Reserved
(1D4)	FULLWORD	4	*	Reserved
(1D8)	FULLWORD	4	*	Reserved
(1DC)	CHARACTER		*	

Array of headers for SUA pool chains.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	SMA_SUA_ ARRAY_POOLHEAD (0 1)	
(0)	CHARACTER	8	SMA_SUA_POOLHEAD	

PPA - Page Pool control Area
There is a PPA for each DSA (ie there are 8). The PPAs are chained from the SMA. In addition there is an array in the SMA which allows each PPA to be addressed directly.
Each SCA contains the address of the PPA from which that subpool is allocated.
Other blocks chained from the PPA are:
PPA_ NEXT - address of next PPA.
PPA_ PREV - address of previous PPA.
PPA_ PPX_FIRST - address of the first PPX for this DSA.
PPA_ PPX_LAST - address of the last PPX for this DSA.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	224	PPA	
(0)	CHARACTER	24	PPA_PREFIX	
(0)	HALFWORD	2	PPA_LENGTH	
(2)	CHARACTER	1	PPA_ARROW	
(3)	CHARACTER	3	PPA_DFH	
(6)	CHARACTER	2	PPA_DOMID	
(8)	CHARACTER	8	PPA_BLOCK_NAME	
(10)	CHARACTER	8	PPA_DSA_NAME	DSA name
(18)	CHARACTER	200	*	
(18)	ADDRESS	4	PPA_NEXT	-> next PPA
(1C)	ADDRESS	4	PPA_PREV	-> previous PPA
(20)	CHARACTER	8	*	
(20)	ADDRESS	4	PPA_PPX_FIRST	-> first PPX
(24)	ADDRESS	4	PPA_PPX_LAST	-> last PPX
(28)	FULLWORD	4	PPA_PAGESIZE	pagesize
(2C)	FULLWORD	4	PPA_PAGEROUND	pagesize rounding value

Offset Hex	Type	Len	Name (Dim)	Description
(30)	FULLWORD	4	PPA_PRIMARY_EXTENT_SIZE	size of primary extent
(34)	FULLWORD	4	PPA_EXTENT_ MULTIPLE	extent multiple value
(38)	FULLWORD	4	PPA_EXTENT_ ROUND	extent rounding value
(3C)	FULLWORD	4	PPA_BOUNDARY	boundary for extents
(40)	FULLWORD	4	PPA_FREE_BYTES	number of free bytes
(44)	FULLWORD	4	PPA_CUSHION_ SIZE	size of cushion
(48)	FULLWORD	4	PPA_LAST_NOTIFY_FREE_BYTES	bytes free last notify
(4C)	FULLWORD	4	PPA_LWM_ FREE_BYTES	low water mark free bytes
(50)	FULLWORD	4	PPA_LARGEST_FREE_AREA	size of largest free area
(54)	FULLWORD	4	PPA_SUSPENDS	number of suspends
(58)	FULLWORD	4	PPA_SUSPENDED	number of tasks suspended
(5C)	FULLWORD	4	PPA_HWM_ SUSPENDED	hwm tasks suspended
(60)	FULLWORD	4	PPA_RESUMED	number resumed
(64)	FULLWORD	4	PPA_REQUESTS_PURGED	number purged
(68)	BITSTRING	1	PPA_FLAGS	
	1...		PPA_SOS	='1'B, currently sos
	.1..		PPA_CUSHION_RELEASED	='1'B, cushion released
	..1.		PPA_ANY	='1'B, DSA is > 16MB
	...1 1111		*	reserved
(69)	UNSIGNED	1	PPA_ACCESS	CICS/USER/READ_ONLY
(6A)	UNSIGNED	1	PPA_INDEX	CDSA, UDSA etc.
(6B)	UNSIGNED	1	*	reserved
(6C)	FULLWORD	4	PPA_DOMAIN_ GETMAINS	getmains for domain subpools already deleted
(70)	FULLWORD	4	PPA_DOMAIN_FREEMAINS	freemains for domain subpools already deleted
(74)	FULLWORD	4	PPA_TASK_ GETMAINS	getmains for task subpools already deleted
(78)	FULLWORD	4	PPA_TASK_ FREEMAINS	freemains for task subpools already deleted
(7C)	FULLWORD	4	PPA_TASK_HWM_PG_STG	HWM for total system task subpool page storage
(80)	FULLWORD	4	PPA_TASK_ CUR_PG_STG	Current total system task subpool page storage
(84)	FULLWORD	4	PPA_ADD_ SUBPOOLS	add_subpool requests
(88)	FULLWORD	4	PPA_DELETE_ SUBPOOLS	delete_subpool requests
(8C)	FULLWORD	4	PPA_GETMAINS_ NOSTG	getmains returning nostg
(90)	FULLWORD	4	PPA_CUSHION_RELEASES	times cushion released
(94)	FULLWORD	4	PPA_TIMES_ WENT_SOS	times went SOS
(98)	CHARACTER	8	PPA_TIME_AT_SOS	total time at SOS
(A0)	FULLWORD	4	PPA_HWM_ FREE_BYTES	high water mark free bytes
(A4)	FULLWORD	4	PPA_STORAGE_VIOLATIONS	number of stg violations
(A8)	CHARACTER	8	PPA_TIME_ WENT_SOS	time last went SOS
(B0)	FULLWORD	4	PPA_NOTIFY_THRESHOLD	threshold for notifies
(B4)	FULLWORD	4	PPA_SIZE	total size
(B8)	ADDRESS	4	PPA_FREEHEAD	free storage header
(BC)	FULLWORD	4	PPA_HWM_SIZE	HWM total size
(C0)	FULLWORD	4	PPA_LWM_SIZE	LWM total size
(C4)	FULLWORD	4	PPA_EXTENTS	number of extents
(C8)	FULLWORD	4	PPA_EXTENTS_ ADDED	extents added
(CC)	FULLWORD	4	PPA_EXTENTS_RELEASED	extents released
(D0)	FULLWORD	4	PPA_REQUESTED_CUSHION_SIZE	cushion size, passed on ADD_DSA call
(D4)	FULLWORD	4	PPA_PAGESIZE_ SHIFT	shift value for pagesize
(D8)	FULLWORD	4	*	reserved
(DC)	FULLWORD	4	*	reserved
(E0)	CHARACTER		*	

PPX - Page Pool extent control area.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	80	PPX	
(0)	CHARACTER	32	PPX_PREFIX	
(0)	HALFWORD	2	PPX_LENGTH	
(2)	CHARACTER	1	PPX_ARROW	
(3)	CHARACTER	3	PPX_DFH	
(6)	CHARACTER	2	PPX_DOMID	
(8)	CHARACTER	8	PPX_BLOCK_NAME	
(10)	CHARACTER	8	PPX_DSA_NAME	DSA name
(18)	ADDRESS	4	PPX_NEXT	-> next PPX
(1C)	ADDRESS	4	PPX_PREV	-> previous PPX
(20)	CHARACTER	48	*	
(20)	FULLWORD	4	PPX_EXTENT_SIZE	size of extent
(24)	ADDRESS	4	PPX_EXTENT_ START	-> start of extent
(28)	ADDRESS	4	PPX_EXTENT_END	-> last byte of extent
(2C)	ADDRESS	4	PPX_SAEP	-> first SAE for extent
(30)	BITSTRING	1	PPX_FLAGS	
	1...		PPX_PRIMARY	='1'B, primary extent
	.111 1111		*	reserved
(31)	CHARACTER	3	*	reserved
(34)	ADDRESS	4	PPX_PAMP	-> start of PAM
(38)	FULLWORD	4	PPX_PAM_BYTES	length of PAM
(3C)	ADDRESS	4	PPX_PPAP	-> PPA
(40)	FULLWORD	4	PPX_FREE_BYTES	free bytes in this extent
(44)	FULLWORD	4	*	reserved
(48)	FULLWORD	4	*	reserved
(4C)	FULLWORD	4	*	reserved
(50)	CHARACTER		*	
(50)	CHARACTER		PPX_PAM_START	page allocation map start

SAT - Storage access table.
Note also that this declaration must be kept in step with the
corresponding declartion in DFHSMSRI.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16928	SAT	
(0)	CHARACTER	16	SAT_PREFIX	eyecatcher
(0)	HALFWORD	2	SAT_LENGTH	
(2)	CHARACTER	1	SAT_ARROW	
(3)	CHARACTER	3	SAT_DFH	
(6)	CHARACTER	2	SAT_DOMID	
(8)	CHARACTER	8	SAT_BLOCK_NAME	
(10)	ADDRESS	4	SAT_BELOWP	-> below vector
(14)	FULLWORD	4	SAT_BELOW_SHIFT	shift for below vector
(18)	ADDRESS	4	SAT_ABOVEP	-> above vector
(1C)	FULLWORD	4	SAT_ABOVE_SHIFT	shift for above vector
(20)	CHARACTER	8	SAT_BELOW (64)	
(220)	CHARACTER	8	SAT_ABOVE (2048)	
(4220)	CHARACTER		*	

SAE - Storage access table entry.
Note that sae_ access and sae_ dsa_name overlay sae_extent_end.
Whenever sae_ extent is used, the second halfword must be set to
zero.
Note also that this declaration must be kept in step with the
corresponding declartion in DFHSMSRI.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	SAE	
(0)	ADDRESS	4	SAE_PPXP	-> PPX
(4)	ADDRESS	4	SAE_EXTENT_END	-> (end of extent)+1
(4)	CHARACTER	2	*	
(6)	UNSIGNED	1	SAE_ACCESS	access value
(7)	UNSIGNED	1	SAE_DSA_NAME	DSA name

CTN - Cartesian Tree Node.
There is a CTN for each node in the cartesian tree structure
which is used to manage free storage for a DSA.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	CTN	
(0)	ADDRESS	4	CTN_LEFT	-> left son/daughter
(4)	ADDRESS	4	CTN_RIGHT	-> right son/daughter
(8)	ADDRESS	4	CTN_ADDR	address of storage area
(C)	UNSIGNED	4	CTN_LEN	length of storage area
(10)	ADDRESS	4	CTN_PPXP	-> PPX for extent
(14)	ADDRESS	4	*	reserved

SMX - Transaction Storage Area.
There is an SMX for each task in the system, excluding true
system tasks ie tasks with no TCA.
Data associated with the task is saved in the SMX, such as the
task lifetime subpool SCA pointers, taskdatakey etc..
The SMXs are chained from the SMA.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	SMX	
(0)	CHARACTER	12	SMX_PREFIX	
(0)	CHARACTER	4	SMX_EYECATCHER	Eyecatcher
(4)	ADDRESS	4	SMX_NEXT	-> Next SMX
(8)	ADDRESS	4	SMX_PREV	-> Previous SMX

==

Do NOT alter the offset of SMX_SUBSPACE_TOKEN, SMX_SUBSPACE_TASK
or SMX_SUBSPACE_ACTIVE without altering DFHSMSRI.

==

Offset Hex	Type	Len	Name (Dim)	Description
(C)	ADDRESS	4	SMX_SUBSPACE_ TOKEN	-> SUA, subspace area
(10)	BITSTRING	1	SMX_FLAGS	Flags
	1...		SMX_CLEAR_STG	='1'B, clear storage on freemaining
	.1..		SMX_FREEZE_STG	='1'B, do not freemain until task end
	..1.		SMX_REMOTE_TRAN	='1'B, task executes remotely
	...1		SMX_ISOLATE	='1'B, task to be isolated from other tasks
	.... 1...		SMX_CICS_ DATAKEY	='1'B, task datakey cics
	.... .1..		SMX_TASKDATALOC_ ANY	='1'B, task dataloc any
	.... ..1.		SMX_SUBSPACE_ TASK	='1'B, task eligible to execute in a subspace
	1		SMX_SUBSPACE_ ACTIVE	='1'B, task is currently executing in a subspace
(11)	CHARACTER	3	*	Reserved
(14)	CHARACTER	4	SMX_TRANSACTION_NUMBER	Transaction number in packed decimal format
(18)	CHARACTER	8	SMX_TRANSACTION_TOKEN	Transaction token

Table of task lifetime subpool SCA pointers.

Offset Hex	Type	Len	Name (Dim)	Description
(20)	CHARACTER	16	SMX_SUBPOOL_TOKEN_TABLE	
(20)	ADDRESS	4	SMX_CICS24_P	-> CICS24 SCA
(24)	ADDRESS	4	SMX_CICS31_P	-> CICS31 SCA
(28)	ADDRESS	4	SMX_USER24_P	-> USER24 SCA
(2C)	ADDRESS	4	SMX_USER31_P	-> USER31 SCA
(30)	CHARACTER	4	*	Reserved
(34)	CHARACTER		*	

SCA - Subpool Control Area.
There is a SCA for each active subpool. Active SCAs are chained
from the SM anchor block. There is also a chain of free SCAs
chained from the SM anchor block.
Other blocks chained from the SCA are:
SCA_ ELEMHEAD - head of the element chain.
SCA_ FREEHEAD - head of the free storage chain.
SCA_ PPAP - address of PPA for this subpool.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	180	SCA	
(0)	CHARACTER	16	SCA_PREFIX	
(0)	CHARACTER	8	SCA_NAME	subpool name
(8)	ADDRESS	4	SCA_NEXT	-> next SCA
(C)	ADDRESS	4	SCA_PREV	-> prev SCA
(10)	CHARACTER	64	*	

The following fields are grouped together as they are referenced
by the in-line macro getmain/free macro DFHSMGFI.
>>>> The offsets of these fields must not be changed without
changing DFHSMGFI also <<<<.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	BITSTRING	1	SCA_FLAGS	flags
	`1... ....`		SCA_QUICKCELL	='1'B, use quickcell
	`.1.. ....`		SCA_INLINE	='1'B, inline code poss
	`..1. ....`		SCA_ANY	='1'B, location(any), ='0'B, location(below)
	`...1 ....`		SCA_RESET_ STATS	='1'B, stats to be reset
	`.... 1...`		SCA_STORAGE_ CHECK	='1'B, storage violation checking for this subpool
	`.... .1..`		SCA_CLEAR_STG	='1'B, clear storage on freemaining
	`.... ..1.`		SCA_FREEZE_STG	='1'B, do not freemain storage until task end
	`.... ...1`		SCA_SELF_ TUNING	='1'B, self-tuning initial-free area
(11)	UNSIGNED	1	SCA_ACCESS	access of DSA in which subpool is allocated
(12)	UNSIGNED	1	SCA_DSA_INDEX	CDSA, UDSA etc.
(13)	CHARACTER	1	*	reserved
(14)	FULLWORD	4	*	reserved
(18)	FULLWORD	4	SCA_FIXEDLEN	fixed length value
(1C)	ADDRESS	4	SCA_FIRST_QPH	-> first QPH
(20)	ADDRESS	4	SCA_LAST_QPH	-> last QPH
(24)	ADDRESS	4	SCA_FIRST_ FREE_QPH	-> first free QPH
(28)	FULLWORD	4	*	reserved
(2C)	HALFWORD	2	SCA_MAX_ FREE_CELLS_LESS1	maximum free cells (less 1).
(2E)	HALFWORD	2	SCA_MIN_ FREE_CELLS	minimum free cells
(30)	FULLWORD	4	SCA_GETMAINS	number of getmains
(34)	ADDRESS	4	SCA_LOCK_TOKEN	subpool lock token
(38)	FULLWORD	4	SCA_FREEMAINS	number of freemains
(3C)	FULLWORD	4	*	reserved
(40)	FULLWORD	4	*	reserved

The following fields are updated by the SM system task for
those subpools which have self-tuning initial-free areas.

Offset Hex	Type	Len	Name (Dim)	Description
(44)	FULLWORD	4	SCA_TUNING_ INTERVALS	self-tuning intervals
(48)	FULLWORD	4	SCA_TUNING_ AVERAGE	tuning average
(4C)	FULLWORD	4	*	reserved
(50)	CHARACTER	100	*	
(50)	CHARACTER	16	SCA_ELEMHEAD	elem chain head
(60)	CHARACTER	16	SCA_FREEHEAD	free chain head
(70)	FULLWORD	4	SCA_NUM	second half of token
(74)	ADDRESS	4	SCA_PPAP	-> Page Pool control Area
(78)	CHARACTER	8	SCA_IFAHEAD	
(78)	ADDRESS	4	SCA_IFA_FIRST	-> first ifa
(7C)	ADDRESS	4	SCA_IFA_LAST	-> last ifa
(80)	FULLWORD	4	SCA_INITFREE_ LEN1	primary ifa size
(84)	FULLWORD	4	SCA_OWNER	owning domain index
(88)	BITSTRING	4	SCA_BDYROUND	boundary mask
(8C)	HALFWORD	2	SCA_BOUNDARY	boundary
(8E)	UNSIGNED	1	SCA_SPID	subpool id
(8F)	UNSIGNED	1	SCA_USAGE	usage
(90)	UNSIGNED	1	SCA_ELEMCHAIN	elemchain option
(91)	UNSIGNED	1	SCA_ELEMTYPE	element type
(92)	CHARACTER	2	*	reserved
(94)	FULLWORD	4	SCA_INITFREE_ LEN2	secondary ifa size
(98)	FULLWORD	4	SCA_PAGE_ STORAGE	page storage
(9C)	FULLWORD	4	SCA_ELEMENT_ STORAGE	element storage (vble only)
(A0)	FULLWORD	4	SCA_NUMELEMS_ LAST_RESET	number of elements at last statistics reset time
(A4)	FULLWORD	4	SCA_HWM_ PAGE_STORG	Subpool HWM page stg
(A8)	ADDRESS	4	SCA_SMXP	-> SMX
(AC)	ADDRESS	4	SCA_SUBSPACE_ TOKEN	-> SUA
(B0)	FULLWORD	4	*	reserved
(B4)	CHARACTER		*	

IFA - initial-free area descriptor.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	IFA	
(0)	ADDRESS	4	IFA_NEXT	-> next ifa
(4)	ADDRESS	4	IFA_PREV	-> previous ifa
(8)	ADDRESS	4	IFA_START	-> area start
(C)	ADDRESS	4	IFA_END	-> area end (last byte+1)
(10)	FULLWORD	4	IFA_LENGTH	length of area
(14)	FULLWORD	4	*	reserved
(18)	CHARACTER		*	

SPC - subpool catalog record.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	16	SPC	
(0)	FULLWORD	4	SPC_TUNING_ INTERVALS	no. of tuning intervals
(4)	FULLWORD	4	SPC_TUNING_ AVERAGE	tuning average
(8)	FULLWORD	4	*	reserved
(C)	FULLWORD	4	*	reserved

SUA - Subspace area.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	SUA	
(0)	CHARACTER	20	SUA_PREFIX	
(0)	CHARACTER	4	SUA_EYECATCHER	Eyecatcher
(4)	CHARACTER	8	SUA_POOL_OR_ALLOC_CHAIN	Pool or alloc chain ptrs
(4)	ADDRESS	4	SUA_NEXT	-> next SUA
(8)	ADDRESS	4	SUA_PREV	-> previous SUA
(C)	ADDRESS	4	SUA_STEAL_NEXT	-> next SUA on the steal chain
(10)	ADDRESS	4	SUA_STEAL_PREV	-> previous SUA on the steal chain

```
==============================================================
Do NOT change the offsets of SUA_QR_ALET or SUA_OPEN_ALET
without altering DFHSMSRI.
==============================================================
```

Offset Hex	Type	Len	Name (Dim)	Description
(14)	UNSIGNED	4	SUA_QR_ALET	Suspace ALET (QR TCB)
(18)	UNSIGNED	4	SUA_OPEN_ALET	Suspace ALET (open TCBS)
(1C)	CHARACTER	8	SUA_STOKEN	Subspace STOKEN
(24)	CHARACTER	8	SUA_SUBSPACE_ NAME	MVS assigned name
(2C)	ADDRESS	4	SUA_TASK_TOKEN	-> SMX
(30)	UNSIGNED	4	SUA_POOL_INDEX	index for pool chains
(34)	BITSTRING	1	SUA_FLAGS	
	`1... ....`		SUA_ALLOCATED_TO_TASK	'1' SUA on the allocated chain
	`.111 1111`		*	Reserved
(35)	CHARACTER	3	*	Reserved
(38)	CHARACTER		*	

SCB - SCA/SCQ/SQE block header.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	SCB	
(0)	CHARACTER	32	SCB_PREFIX	
(0)	HALFWORD	2	SCB_LENGTH	
(2)	CHARACTER	1	SCB_ARROW	
(3)	CHARACTER	3	SCB_DFH	
(6)	CHARACTER	2	SCB_DOMID	
(8)	CHARACTER	8	SCB_BLOCK_NAME	
(10)	ADDRESS	4	SCB_NEXT	-> next SCB
(14)	ADDRESS	4	*	reserved
(18)	ADDRESS	4	*	reserved
(1C)	ADDRESS	4	*	reserved
(20)	CHARACTER		*	

QPH - Quickcell page header block.
Note that offsets must remain the same as within the inline
getmain/freemain macro DFHSMGFI.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	48	QPH	
(0)	CHARACTER	32	QPH_PREFIX	
(0)	HALFWORD	2	QPH_LENGTH	
(2)	CHARACTER	1	QPH_ARROW	
(3)	CHARACTER	3	QPH_DFH	
(6)	CHARACTER	2	QPH_DOMID	
(8)	CHARACTER	8	QPH_BLOCK_NAME	
(10)	CHARACTER	8	QPH_NAME	subpool name
(18)	ADDRESS	4	QPH_NEXT	-> next QPH
(1C)	ADDRESS	4	QPH_PREV	-> previous QPH
(20)	CHARACTER	16	*	
(20)	ADDRESS	4	QPH_NEXT_FREE	-> next QPH on free chain
(24)	ADDRESS	4	QPH_FIRST_ FREE_CELL	-> first free cell
(28)	HALFWORD	2	QPH_NUMBER_ FREE_CELLS	current free cells
(2A)	CHARACTER	2	QPH_FLAGS	
(2A)	BITSTRING	1	*	
	1...		QPH_DONT_ FREE_PAGE	='1'b, don't free page when empty
	.1..		QPH_ON_ FREE_CHAIN	='1'B, page is on free chain
	..11 1111		*	reserved
(2B)	BITSTRING	1	*	reserved
(2C)	ADDRESS	4	QPH_SCAP	-> SCA owning subpool
(30)	CHARACTER		*	

QPF - quickcell page free element.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	QPF	
(0)	ADDRESS	4	QPF_SCAP	free element check field
(4)	ADDRESS	4	QPF_NEXT	-> next quickcell element

SCQ - quickcell element (for SCE and SCF descriptors)

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	SCQ	
(0)	ADDRESS	4	SCQ_NEXT	-> next quickcell element

SCE - element descriptor

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	SCE	
(0)	CHARACTER	16	SCE_PREFIX	
(0)	ADDRESS	4	SCE_NEXT	-> next element descriptor
(4)	ADDRESS	4	SCE_PREV	-> prev element descriptor
(8)	ADDRESS	4	SCE_ADDR	-> element storage
(C)	FULLWORD	4	SCE_LEN	element length
(10)	CHARACTER	8	*	
(10)	ADDRESS	4	SCE_PPXP	-> PPX
(14)	ADDRESS	4	*	reserved
(18)	CHARACTER		*	

SCF - free storage descriptor.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	SCF	
(0)	CHARACTER	16	SCF_PREFIX	
(0)	ADDRESS	4	SCF_NEXT	-> next SCF
(4)	ADDRESS	4	SCF_PREV	-> previous SCF
(8)	ADDRESS	4	SCF_ADDR	-> free storage block
(C)	FULLWORD	4	SCF_LEN	free storage length
(10)	CHARACTER	8	*	
(10)	ADDRESS	4	SCF_PPXP	-> PPX
(14)	ADDRESS	4	*	reserved
(18)	CHARACTER		*	

SQE - suspend queue element.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	52	SQE	
(0)	ADDRESS	4	SQE_NEXT	-> next SQE
(4)	ADDRESS	4	SQE_PREV	-> previous SQE
(8)	ADDRESS	4	SQE_SCAP	-> SCA
(C)	FULLWORD	4	SQE_BYTES_ REQUESTED	requested bytes
(10)	ADDRESS	4	SQE_SUSPEND_ TOKEN	DS suspend token
(14)	ADDRESS	4	SQE_TASK_TOKEN	KE task token
(18)	CHARACTER	8	SQE_SUSPEND_ START	time suspend issued
(20)	ADDRESS	4	*	Reserved
(24)	CHARACTER	4	SQE_TRANSACTION_ NUMBER	
(28)	BITSTRING	1	SQE_FLAGS	
	1...		SQE_DELETED	logically deleted
	.111 1111		*	reserved
(29)	CHARACTER	3	*	reserved
(2C)	FULLWORD	4	*	reserved
(30)	FULLWORD	4	*	reserved
(34)	CHARACTER		*	

DXH - DSA extent list header.
Note: DXH/DXE declarations must be kept in step with those in
DFHSMAFI.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	328	DXH	
(0)	CHARACTER	16	DXH_PREFIX	
(0)	HALFWORD	2	DXH_LENGTH	
(2)	CHARACTER	1	DXH_ARROW	
(3)	CHARACTER	3	DXH_DFH	
(6)	CHARACTER	2	DXH_DOMID	
(8)	CHARACTER	8	DXH_BLOCK_NAME	
(10)	CHARACTER	16	*	
(10)	BITSTRING	1	DXH_FLAGS	
	1...		DXH_STORAGE_ PROTECT	
	.1..		DXH_REENTRANT_ PROGRAM_PROTECT	
	..1.		DXH_TRANSACTION_ ISOLATION	
	...1		DXH_LOC_ EXPLICIT	
	 1111		*	
(11)	CHARACTER	3	*	
(14)	ADDRESS	4	DXH_FREE_HEAD	
(18)	FULLWORD	4	DXH_EXTENT_ MULTIPLE_BELOW	
(1C)	FULLWORD	4	DXH_EXTENT_ MULTIPLE_ABOVE	
(20)	CHARACTER	120	*	
(20)	CHARACTER	40	DXH_BELOW_ GETMAIN_HEAD	
(48)	CHARACTER	40	DXH_BELOW_ EXTENT_HEAD	

Offset Hex	Type	Len	Name (Dim)	Description
(70)	CHARACTER	40	DXH_BELOW_ LD_CHECK_HEAD	
(98)	CHARACTER	120	*	
(98)	CHARACTER	40	DXH_ABOVE_ GETMAIN_HEAD	
(C0)	CHARACTER	40	DXH_ABOVE_ EXTENT_HEAD	
(E8)	CHARACTER	40	DXH_ABOVE_ LD_CHECK_HEAD	
(110)	CHARACTER	56	*	
(110)	ADDRESS	4	DXH_TRACEP	
(114)	ADDRESS	4	DXH_VGETSP	
(118)	UNSIGNED	4	DXH_GET_ DSALIM_REQUESTS	
(11C)	UNSIGNED	4	DXH_GET_ DSALIM_ REQUESTS_NOSTG	
(120)	UNSIGNED	4	DXH_ALLOCATE_ DSA_EXTENT_ REQUESTS	
(124)	UNSIGNED	4	DXH_EXTENT_ GETMAINS	
(128)	UNSIGNED	4	DXH_EXTENT_ GETMAINS_EXPLICIT	
(12C)	UNSIGNED	4	DXH_EXTENT_ GETMAINS_SINGLE	
(130)	UNSIGNED	4	DXH_EXTENT_ GETMAINS_VTYPE	
(134)	UNSIGNED	4	DXH_EXTENT_ GETMAINS_NOSTG	
(138)	FULLWORD	4	*	reserved
(13C)	FULLWORD	4	*	reserved
(140)	FULLWORD	4	*	reserved
(144)	FULLWORD	4	*	reserved
(148)	CHARACTER		*	

DXG - DSA extent getmain descriptor.
Note: Next/prev pointers in must be at the same offset as in DXE.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	40	DXG	
(0)	ADDRESS	4	DXG_NEXT	-> next DXG
(4)	ADDRESS	4	DXG_PREV	-> previous DXG
(8)	ADDRESS	4	DXG_ADDR	address of getmained area
(C)	ADDRESS	4	DXG_LEN	length of getmained area
(10)	UNSIGNED	1	DXG_MVS_SUBPOOL	MVS subpool of extent
(11)	UNSIGNED	1	DXG_MVS_KEY	MVS storage key of extent
(12)	CHARACTER	2	*	reserved
(14)	FULLWORD	4	*	reserved

DXE - DSA extent list element.
Notes:
1. DXH/DXE declarations must be kept in step with those in DFHSMAFI.
2. Next/prev pointers in must be at the same offset as in DXG.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	40	DXE	
(0)	ADDRESS	4	DXE_NEXT	-> next DXE
(4)	ADDRESS	4	DXE_PREV	-> previous DXE
(8)	ADDRESS	4	DXE_LD_ CHECK_NEXT	-> next LD check DXE
(C)	ADDRESS	4	DXE_LD_ CHECK_PREV	-> previous LD check DXE
(10)	ADDRESS	4	DXE_EXTENT_START	-> start of extent
(14)	ADDRESS	4	DXE_EXTENT_END	-> end of extent
(18)	ADDRESS	4	DXE_DXGP	-> "owning" DXG
(1C)	ADDRESS	4	DXE_PPXP	-> PPX for extent
(20)	BITSTRING	1	DXE_FLAGS	flags
	1...		DXE_IDENTIFIED	='1'b, extent identify'd
	.111 1111		*	reserved
(21)	UNSIGNED	1	DXE_DSA_NAME	DSA index of extent
(22)	CHARACTER	2	*	reserved
(24)	FULLWORD	4	*	reserved

Catalog record.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	CAT	
(0)	BITSTRING	4	CAT_FLAGS	
(0)	BITSTRING	1	*	
	1...		CAT_STORAGE_PROTECT_REQ	stgprot reqd
	.1..		CAT_TRAN_ISOLATION_REQ	traniso reqd
	..11 1111		*	reserved
(1)	BITSTRING	3	*	reserved
(4)	CHARACTER	8	*	
(4)	UNSIGNED	4	CAT_DSA_LIMIT	
(8)	UNSIGNED	4	CAT_EDSA_LIMIT	
(C)	CHARACTER		*	

Constants

Len	Type	Value	Name	Description
4	CHARACTER	>SMX	SMX_NAME	Eyecatcher
8	CHARACTER	SMSUBPOL	SPC_TYPE	
Subpool name in SCA header block.				
8	CHARACTER	(HEADER)	SCA_HEAD_NAME	
8	CHARACTER	(FREE)	SCA_FREE_NAME	
4	CHARACTER	>SUA	SUA_NAME	Eyecatcher
Block names for above.				
8	CHARACTER	DXEBLOCK	DXEBLOCK_NAME	
8	CHARACTER	SATBLOCK	SATBLOCK_NAME	
8	CHARACTER	SCABLOCK	SCABLOCK_NAME	
8	CHARACTER	SCQBLOCK	SCQBLOCK_NAME	
8	CHARACTER	SQEBLOCK	SQEBLOCK_NAME	
8	CHARACTER	SMXBLOCK	SMXBLOCK_NAME	
8	CHARACTER	SUABLOCK	SUABLOCK_NAME	
8	CHARACTER	SMDOMAIN	CAT_TYPE	
8	CHARACTER	SMSTATE	CAT_NAME	
Miscellaneous constants.				
1	CHARACTER	>	ARROW	
4	DECIMAL	8	BDY8	
4	DECIMAL	16	BDY16	
4	HEX	FFFFFFF0	BDY16ROUND	
4	DECIMAL	32	BDY32	
4	HEX	FFFFFFE0	BDY32ROUND	
4	DECIMAL	255	SYSTEM_TASK_ PRIORITY	
8	CHARACTER	SMSYSTEM	SYSTEM_TASK_SUSPEND_NAME	
4	DECIMAL	60	SYSTEM_TASK_SUSPEND_INTERVAL	
4	DECIMAL	2	SYSTEM_TASK_SUSPEND_INTERVAL_ SOS	
4	DECIMAL	16777216	MB16	
8	CHARACTER	SMLOCK	SMLOCK_NAME	
4	HEX	7FFFFFFF	SCF_NULL	
4	DECIMAL	16384	BYTES_FOR_ABENDING_TASKS	
4	DECIMAL	100	MXT_ADJUSTMENT_	
4	DECIMAL	128	STORAGE_VIOLATION_DATA_LEN	
Pre-allocated subpool id's.				
4	DECIMAL	0	SPID_FREE	free page
4	DECIMAL	1	SPID_TASK_CICS24	CICS24 spid
4	DECIMAL	2	SPID_TASK_USER24	USER24 spid
4	DECIMAL	3	SPID_TASK_CICS31	CICS31 spid
4	DECIMAL	4	SPID_TASK_USER31	USER31 spid
4	DECIMAL	5	SPID_DOMAIN_FIRST	first domain spid
Prefixes for task subpool names.				
1	CHARACTER	M	PREF_TASK_CICS24	
1	CHARACTER	B	PREF_TASK_USER24	
1	CHARACTER	C	PREF_TASK_CICS31	
1	CHARACTER	U	PREF_TASK_USER31	
Trace point id's.				
2	HEX	0101	TID_SMDM_ENTRY	

Len	Type	Value	Name	Description
2	HEX	0102	TID_SMDM_EXIT	
2	HEX	0103	TID_SMDM_RECOVERY	
2	HEX	0104	TID_SMDM_NOSTG_SMA	
2	HEX	0109	TID_SMDM_ NOSTG_SCAB	
2	HEX	010A	TID_SMDM_ NOSTG_SCQB	
2	HEX	010C	TID_SMDM_ STCK_ERROR	
2	HEX	010D	TID_SMDM_ NOSTG_STAB	
2	HEX	010E	TID_SMDM_ NOSTG_SMXB	
2	HEX	010F	TID_SMDM_INVALID_FORMAT	
2	HEX	0110	TID_SMDM_INVALID_FUNCTION	
2	HEX	0111	TID_SMDM_NOSTG_REQ_DSALIM	
2	HEX	0112	TID_SMDM_NOSTG_REQ_EDSALIM	
2	HEX	0113	TID_SMDM_NOSTG_DFT_DSALIM	
2	HEX	0114	TID_SMDM_NOSTG_DFT_EDSALIM	
2	HEX	0115	TID_SMDM_ SVC_CALL_FAIL	
2	HEX	0116	TID_SMDM_NOSTG_DSA	
2	HEX	0201	TID_SMAD_ENTRY	
2	HEX	0202	TID_SMAD_EXIT	
2	HEX	0203	TID_SMAD_RECOVERY	
2	HEX	0204	TID_SMAD_INVALID_FORMAT	
2	HEX	0205	TID_SMAD_INVALID_FUNCTION	
2	HEX	0206	TID_SMAD_NO_MVS_STORAGE	
2	HEX	0207	TID_SMAD_SUBPOOL_NOT_EMPTY	
2	HEX	0208	TID_SMAD_INVALID_SUBPOOL_ TOKEN	
2	HEX	0F01	TID_SMAR_ENTRY	
2	HEX	0F02	TID_SMAR_EXIT	
2	HEX	0F03	TID_SMAR_RECOVERY	
2	HEX	0F04	TID_SMAR_INVALID_FORMAT	
2	HEX	0F05	TID_SMAR_INVALID_FUNCTION	
2	HEX	0F06	TID_SMAR_SET_TRAN_TOKEN_ FAIL	
2	HEX	0F07	TID_SMAR_ INQ_TRAN_FAIL	
2	HEX	0F08	TID_SMAR_INQ_TRAN_TOKEN_ FAIL	
2	HEX	0F09	TID_SMAR_NO_MVS_STORAGE_ SCA	
2	HEX	0F0A	TID_SMAR_NO_MVS_STORAGE_ SCQ	
2	HEX	0F0B	TID_SMAR_NO_MVS_STORAGE_ SMX	
2	HEX	0F0C	TID_SMAR_STGCHK_FAILURE	
2	HEX	0F0D	TID_SMAR_FREEMAIN_ELEM	
2	HEX	0F0E	TID_SMAR_STG_VIOL_PCT_INC_ FAIL	
2	HEX	0F0F	TID_SMAR_STG_VIOL_TCT_INC_ FAIL	
2	HEX	0301	TID_SMGF_ENTRY	
2	HEX	0302	TID_SMGF_EXIT	
2	HEX	0303	TID_SMGF_RECOVERY	
2	HEX	0304	TID_SMGF_INVALID_FUNCTION	
2	HEX	0305	TID_SMGF_INVALID_ADDRESS	
2	HEX	0306	TID_SMGF_NO_MVS_STORAGE	
2	HEX	030A	TID_SMGF_INSUFFICIENT_STORAGE	
2	HEX	030B	TID_SMGF_STGCHK_FAILURE	
2	HEX	030C	TID_SMGF_INVALID_INITIAL_ IMAGE	
2	HEX	030D	TID_SMGF_QCELL_GETMAIN_ INV_QPF	
2	HEX	030E	TID_SMGF_QCELL_FREEMAIN_INV_QPH	
2	HEX	030F	TID_SMGF_QCELL_ALREADY_ FREE	
2	HEX	0310	TID_SMGF_QCELL_INV_FREE_ CHAIN	
2	HEX	0311	TID_SMGF_GETMAIN_INV_STG_ CLASS	

Len	Type	Value	Name	Description
2	HEX	0312	TID_SMGF_FREEMAIN_INV_STG_CLASS	
2	HEX	0313	TID_SMGF_GETMAIN_NO_TRAN_ ENV	
2	HEX	0314	TID_SMGF_FREEMAIN_NO_TRAN_ ENV	
2	HEX	0315	TID_SMGF_INV_ADDR_STG_CLASS	
The following 3 trace pts are reserved for APAR PN24591.				
2	HEX	0316	TID_SMGF_PAGES_NOT_OWNED	
2	HEX	0317	TID_SMGF_NEXT_SCF_OVERLAY	
2	HEX	0318	TID_SMGF_PREV_SCF_OVERLAY	
2	HEX	0319	TID_SMGF_STG_VIOL_PCT_INC_ FAIL	
2	HEX	031A	TID_SMGF_STG_VIOL_TCT_INC_ FAIL	
2	HEX	031B	TID_SMGF_NO_MVS_STORAGE_ SQE	
2	HEX	031C	TID_SMGF_ STG_FREEZE	
2	HEX	031D	TID_SMGF_QCELL_SCAP_FOUND	
2	HEX	031E	TID_SMGF_SUBPOOL_LOCK_FAILED	
2	HEX	031F	TID_SMGF_SUBPOOL_UNLOCK_ FAILED	
2	HEX	0401	TID_SMSR_ENTRY	
2	HEX	0402	TID_SMSR_EXIT	
2	HEX	0403	TID_SMSR_RECOVERY	
2	HEX	0404	TID_SMSR_INVALID_FORMAT	
2	HEX	0405	TID_SMSR_INVALID_FUNCTION	
2	HEX	0406	TID_SMSR_ LOCK_ERROR	
2	HEX	0407	TID_SMSR_UNLOCK_ERROR	
2	HEX	0601	TID_SMMCI_ENTRY	
2	HEX	0602	TID_SMMCI_EXIT	
2	HEX	0603	TID_SMMCI_RECOVERY	
2	HEX	0801	TID_SMSY_ENTRY	
2	HEX	0802	TID_SMSY_EXIT	
2	HEX	0803	TID_SMSY_RECOVERY	
2	HEX	0804	TID_SMSY_INVALID_FORMAT	
2	HEX	0805	TID_SMSY_INVALID_FUNCTION	
2	HEX	0808	TID_SMSY_BEFORE_SUSPEND	
2	HEX	0809	TID_SMSY_ AFTER_RESUME	
2	HEX	080A	TID_SMSY_SOS	
2	HEX	080B	TID_SMSY_NOT_SOS	
2	HEX	080C	TID_SMSY_ INVALID_STATE	
2	HEX	0901	TID_SMCK_ENTRY	
2	HEX	0902	TID_SMCK_EXIT	
2	HEX	0903	TID_SMCK_RECOVERY	
2	HEX	0904	TID_SMCK_INVALID_FORMAT	
2	HEX	0905	TID_SMCK_INVALID_FUNCTION	
2	HEX	0906	TID_SMCK_ LOCK_ERROR	
2	HEX	0907	TID_SMCK_UNLOCK_ERROR	
2	HEX	090A	TID_SMCK_SAACHK_TP	
2	HEX	0910	TID_SMCK_ SAA_NOT_BDY8	
2	HEX	0911	TID_SMCK_SAA_NOT_IN_DSA	
2	HEX	0912	TID_SMCK_SAA_INV_SUBPOOL_ ID	
2	HEX	0913	TID_SMCK_SAA_LENGTH_ZERO	
2	HEX	0914	TID_SMCK_SAA_LENGTH_NOT_ MULT8	
2	HEX	0915	TID_SMCK_DUP_SAA_NOT_IN_ DSA	
2	HEX	0916	TID_SMCK_SAA_LENGTH_INVALID	
2	HEX	0917	TID_SMCK_SAA_CLASS_INVALID	
2	HEX	0930	TID_SMCK_SAA_RECOVERED	
2	HEX	0931	TID_SMCK_TCTTE_RECOVERED	

Len	Type	Value	Name	Description
2	HEX	0932	TID_SMCK_ ZONE_CHECK_FAILED	
2	HEX	0933	TID_SMCK_ TIOA_CHAIN_LOOP	
2	HEX	0934	TID_SMCK_ ZONES_RECOVERED	
2	HEX	0935	TID_SMCK_ STG_VIOL_PCT_INC_ FAIL	
2	HEX	0936	TID_SMCK_ STG_VIOL_TCT_INC_ FAIL	
2	HEX	0937	TID_SMCK_ SWITCH_TO_QR_FAIL	
2	HEX	0938	TID_SMCK_ SWITCH_FROM_QR_ FAIL	
2	HEX	0A01	TID_SMST_ENTRY	
2	HEX	0A02	TID_SMST_EXIT	
2	HEX	0A03	TID_SMST_RECOVERY	
2	HEX	0A04	TID_SMST_ INVALID_FORMAT	
2	HEX	0A05	TID_SMST_ INVALID_FUNCTION	
2	HEX	0A06	TID_SMST_ INVALID_PARAMETERS	
2	HEX	0A07	TID_SMST_ LOCK_ERROR	
2	HEX	0A08	TID_SMST_ UNLOCK_ERROR	
2	HEX	0A09	TID_SMST_ INVALID_BUFFER	
2	HEX	0C01	TID_SMMG_ENTRY	
2	HEX	0C02	TID_SMMG_EXIT	
2	HEX	0C03	TID_SMMG_RECOVERY	
2	HEX	0C04	TID_SMMG_ NO_TCTTE_ADDRESS	
2	HEX	0C05	TID_SMMG_ INV_STORAGE_CLASS	
2	HEX	0C06	TID_SMMG_ CICS24_INV_GET_ LENGTH	
2	HEX	0C08	TID_SMMG_ SHRC24_INV_GET_ LENGTH	
2	HEX	0C09	TID_SMMG_ TP_INV_GET_LENGTH	
2	HEX	0C0A	TID_SMMG_ NO_MVS_STORAGE	
2	HEX	0C0B	TID_SMMG_ USER24_INV_GET_ LENGTH	
2	HEX	0C0C	TID_SMMG_ INSUFFICIENT_STORAGE	
2	HEX	0C0E	TID_SMMG_ USER31_INV_GET_ LENGTH	
2	HEX	0C11	TID_SMMG_ SHRU24_INV_GET_ LENGTH	
2	HEX	0C12	TID_SMMG_ SHRU31_INV_GET_ LENGTH	
2	HEX	0C13	TID_SMMG_ INVALID_FUNCTION	
2	HEX	0C14	TID_SMMG_ CICS31_INV_GET_ LENGTH	
2	HEX	0C15	TID_SMMG_ SHRC31_INV_GET_ LENGTH	
2	HEX	0C16	TID_SMMG_ TASK_INV_GET_LENGTH	
2	HEX	0C17	TID_SMMG_ TASK24_INV_GET_ LENGTH	
2	HEX	0C18	TID_SMMG_ CICS24_SAA_INV_ GET_LEN	
2	HEX	0C19	TID_SMMG_ SHRC24_SAA_INV_ GET_LEN	
2	HEX	0C1A	TID_SMMG_ NO_TRAN_ENV	
2	HEX	0D01	TID_SMMF_ENTRY	
2	HEX	0D02	TID_SMMF_EXIT	
2	HEX	0D03	TID_SMMF_RECOVERY	
2	HEX	0D05	TID_SMMF_ SAACHK_F_TP	
2	HEX	0D06	TID_SMMF_ ADDR_NOT_BDY8	
2	HEX	0D07	TID_SMMF_ ADDR_OUTSIDE_DSA	
2	HEX	0D08	TID_SMMF_ ADDR_IN_FREE_PAGE	
2	HEX	0D09	TID_SMMF_ NO_TCTTE_ADDRESS	
2	HEX	0D0A	TID_SMMF_ TP_ADDR_NOT_FOUND	
2	HEX	0D0C	TID_SMMF_ INVALID_ADDRESS	
2	HEX	0D0D	TID_SMMF_ NO_MVS_STORAGE	

Len	Type	Value	Name	Description
2	HEX	0D10	TID_SMMF_INVALID_FUNCTION	
2	HEX	0D11	TID_SMMF_STGCHK_FAILURE	
2	HEX	0D12	TID_SMMF_INVALID_EXEC_KEY	
The following 3 trace pts are reserved for APAR PN24591.				
2	HEX	0D13	TID_SMMF_PAGES_NOT_OWNED	
2	HEX	0D14	TID_SMMF_NEXT_SCF_OVERLAY	
2	HEX	0D15	TID_SMMF_PREV_SCF_OVERLAY	
2	HEX	0D16	TID_SMMF_STG_VIOL_PCT_INC_ FAIL	
2	HEX	0D17	TID_SMMF_STG_VIOL_TCT_INC_ FAIL	
2	HEX	0D18	TID_SMMF_ NO_TRAN_ENV	
2	HEX	0D19	TID_SMMF_ STG_FREEZE	
2	HEX	0E01	TID_SMMC2_ENTRY	
2	HEX	0E02	TID_SMMC2_EXIT	
2	HEX	0E03	TID_SMMC2_RECOVERY	
2	HEX	0E04	TID_SMMC2_INVALID_FUNCTION	
2	HEX	0E05	TID_SMMC2_FREEMAIN_ELEM	
2	HEX	0E06	TID_SMMC2_SAACHK_F_ALL_TP	
2	HEX	0E08	TID_SMMC2_NO_MVS_STORAGE	
2	HEX	0E0A	TID_SMMC2_INVALID_ADDRESS	
2	HEX	0E0B	TID_SMMC2_STGCHK_FAILURE	
The following 3 trace pts are reserved for APAR PN24591.				
2	HEX	0E0D	TID_SMMC2_PAGES_NOT_OWNED	
2	HEX	0E0E	TID_SMMC2_NEXT_SCF_OVERLAY	
2	HEX	0E0F	TID_SMMC2_PREV_SCF_OVERLAY	
2	HEX	0E10	TID_SMMC2_STG_VIOL_PCT_INC_ FAIL	
2	HEX	0E11	TID_SMMC2_STG_VIOL_TCT_INC_ FAIL	
2	HEX	0E12	TID_SMMC2_ NO_TRAN_ENV	
2	HEX	1001	TID_SMSQ_ENTRY	
2	HEX	1002	TID_SMSQ_EXIT	
2	HEX	1003	TID_SMSQ_RECOVERY	
2	HEX	1004	TID_SMSQ_INVALID_FORMAT	
2	HEX	1005	TID_SMSQ_INVALID_FUNCTION	
2	HEX	1006	TID_SMSQ_DSSR_INQUIRE_SUSPEND	
2	HEX	1007	TID_SMSQ_BEFORE_SUSPEND	
2	HEX	1008	TID_SMSQ_AFTER_SUSPEND	
2	HEX	1009	TID_SMSQ_NO_MVS_STORAGE_ SQE	
2	HEX	1101	TID_SMPP_ENTRY	
2	HEX	1102	TID_SMPP_EXIT	
2	HEX	1103	TID_SMPP_RECOVERY	
2	HEX	1104	TID_SMPP_INVALID_FORMAT	
2	HEX	1105	TID_SMPP_INVALID_FUNCTION	
2	HEX	1106	TID_SMPP_NOSTG_PPA	
2	HEX	1107	TID_SMPP_NOSTG_PPX	
2	HEX	1109	TID_SMPP_NOSTG_SAT	
2	HEX	110D	TID_SMPP_NOSTG_CTN	
2	HEX	110E	TID_SMPP_DELETING_EMPTY_ EXTENT	
2	HEX	110F	TID_SMPP_BEFORE_SVC_CALL	
2	HEX	1110	TID_SMPP_AFTER_SVC_CALL	
2	HEX	1111	TID_SMPP_FREE_DSA_LIMIT_ FAILED	
2	HEX	1112	TID_SMPP_ SVC_CALL_FAIL	
2	HEX	1113	TID_SMPP_ALLOCATE_EXTENT_ FAILED	
2	HEX	1201	TID_SMPQ_ENTRY	
2	HEX	1202	TID_SMPQ_EXIT	

Len	Type	Value	Name	Description
2	HEX	1203	TID_SMPQ_RECOVERY	
2	HEX	1204	TID_SMPQ_INVALID_FORMAT	
2	HEX	1205	TID_SMPQ_INVALID_FUNCTION	
2	HEX	1206	TID_SMPQ_INSUFFICIENT_STORAGE	
2	HEX	1207	TID_SMPQ_INVALID_ADDRESS	
2	HEX	1208	TID_SMPQ_NOSTG_CTN	
2	HEX	1209	TID_SMPQ_BEFORE_SVC_CALL	
2	HEX	120A	TID_SMPQ_AFTER_SVC_CALL	
2	HEX	120B	TID_SMPQ_ SVC_CALL_FAIL	
2	HEX	3001	TID_SMSU_ENTRY	
2	HEX	3002	TID_SMSU_EXIT	
2	HEX	3003	TID_SMSU_RECOVERY	
2	HEX	3004	TID_SMSU_INVALID_FUNCTION	
2	HEX	3005	TID_SMSU_CHANGE_MODE_FAIL1	
2	HEX	3006	TID_SMSU_SUA_MVS_GETMAIN_ FAIL	
2	HEX	3007	TID_SMSU_ALESERV_ADD_FAIL_ ALLOC	
2	HEX	3008	TID_SMSU_WRONG_TCB_FOR_ALLOCATE	
2	HEX	3009	TID_SMSU_CREATE_SUBSPACE_ENTRY	
2	HEX	300A	TID_SMSU_CREATE_SUBSPACE_ EXIT	
2	HEX	300B	TID_SMSU_IARSUBSP_CREATE_ FAIL	
2	HEX	300C	TID_SMSU_WRONG_TCB_FOR_ DELETE	
2	HEX	300D	TID_SMSU_DELETE_SUBSPACE_ENTRY	
2	HEX	300E	TID_SMSU_DELETE_SUBSPACE_ EXIT	
2	HEX	300F	TID_SMSU_IARSUBSP_DELETE_ FAIL	
2	HEX	3010	TID_SMSU_BAD_PAGE_MULTIPLE	
2	HEX	3011	TID_SMSU_IARSUBSP_ASSIGN_ FAIL	
2	HEX	3012	TID_SMSU_BAD_ELEM_ALIGN	
2	HEX	3013	TID_SMSU_ INVALID_INPUT_SPACE	
2	HEX	3014	TID_SMSU_ALESERV_ADD_FAIL_ STEAL	
2	HEX	3016	TID_SMSU_ALESERV_DELETE_ FAIL	
2	HEX	3018	TID_SMSU_ ALET_STEAL	
2	HEX	3019	TID_SMSU_IARSUBSP_UNASSIGN_ FAIL	
2	HEX	301B	TID_SMSU_INVALID_FORMAT	
2	HEX	301C	TID_SMSU_ ASSIGN_ENTRY	
2	HEX	301D	TID_SMSU_ ASSIGN_EXIT	
2	HEX	301E	TID_SMSU_UNASSIGN_ENTRY	
2	HEX	301F	TID_SMSU_ UNASSIGN_EXIT	
2	HEX	3020	TID_SMSU_CHANGE_MODE_FAIL2	
2	HEX	3021	TID_SMSU_WRONG_TCB_FOR_RELEASE	
2	HEX	3022	TID_SMSU_ASSIGN_FAIL_ABEND	
2	HEX	3023	TID_SMSU_UNASSIGN_FAIL_ ABEND	
2	HEX	3024	TID_SMSU_TEST	
2	HEX	3025	TID_SMSU_NO_ALET_TO_STEAL	
2	HEX	3026	TID_SMSU_ SVC_CALL_FAIL	
2	HEX	3027	TID_SMSU_MULT_UNASSIGN_ ENTRY	
2	HEX	3028	TID_SMSU_FREE_SUBSP_TCBS_ FAIL	

SMSCP point id's are AP domain's.

2	HEX	F101	TID_SMSCP_ENTRY	

Len	Type	Value	Name	Description
2	HEX	F102	TID_SMSCP_EXIT	
2	HEX	F104	TID_SMSCP_INVALID_REQUEST	
Minimum, maximum and default DSALIMIT values				
4	DECIMAL	2097152	MIN_DSA_LIMIT	
4	DECIMAL	16777216	MAX_DSA_LIMIT	
4	DECIMAL	5242880	DEFAULT_DSA_LIMIT	
Minimum, maximum and default EDSALIMIT values				
4	DECIMAL	10485760	MIN_EDSA_LIMIT	
4	DECIMAL	2146435072	MAX_EDSA_LIMIT	2G-1M
4	DECIMAL	20971520	DEFAULT_EDSA_LIMIT	
Multiple for DSA extents (to be kept in step with dsa_ extent_ shift and edsa_extent_shift below).				
4	DECIMAL	262144	DSA_MULTIPLE	
4	DECIMAL	1048576	EDSA_MULTIPLE	
Shift values for use with SAT (to be kept in step with dsa_ multiple and edsa_ multiple above).				
4	DECIMAL	18	DSA_EXTENT_SHIFT	
4	DECIMAL	20	EDSA_EXTENT_SHIFT	
Standard message numbers and system dumpcode values.				
4	DECIMAL	1	MNO_ABEND	
8	CHARACTER	SM0001	DCD_ABEND	
4	DECIMAL	2	MNO_SEVERE_ERROR	
8	CHARACTER	SM0002	DCD_SEVERE_ERROR	
4	DECIMAL	3	MNO_NO_STORAGE	
8	CHARACTER	SM0003	DCD_NO_STORAGE	
4	DECIMAL	4	MNO_LOOP	
8	CHARACTER	SM0004	DCD_LOOP	
4	DECIMAL	5	MNO_STCK_ERROR	
8	CHARACTER	SM0005	DCD_STCK_ERROR	
4	DECIMAL	6	MNO_NO_MVS_STORAGE	
8	CHARACTER	SM0006	DCD_NO_MVS_STORAGE	
Non-standard message numbers and system dumpcode values.				
4	DECIMAL	102	MNO_STORAGE_ VIOLATION	
8	CHARACTER	SM0102	DCD_STORAGE_ VIOLATION	
4	DECIMAL	103	MNO_FAQE_ERROR	
8	CHARACTER	SM0103	DCD_FAQE_ERROR	
4	DECIMAL	113	MNO_NO_STOR_PROT	
4	DECIMAL	114	MNO_STOR_PROT_REQ	
4	DECIMAL	115	MNO_STOR_PROT	
4	DECIMAL	120	MNO_RENTPGM	
4	DECIMAL	122	MNO_DSA_LIMIT	
4	DECIMAL	123	MNO_EDSA_LIMIT	
4	DECIMAL	124	MNO_TRAN_ISO_REQ	
4	DECIMAL	125	MNO_TRAN_ISO	
4	DECIMAL	126	MNO_NO_TRAN_ISO	
4	DECIMAL	127	MNO_NOSTG_ REQ_DSALIM	
4	DECIMAL	128	MNO_NOSTG_REQ_EDSALIM	
4	DECIMAL	129	MNO_NOSTG_ DFT_DSALIM	
4	DECIMAL	130	MNO_NOSTG_DFT_EDSALIM	
4	DECIMAL	131	MNO_SOS_BELOW	
4	DECIMAL	132	MNO_NOT_SOS_BELOW	
4	DECIMAL	133	MNO_SOS_ABOVE	
4	DECIMAL	134	MNO_NOT_SOS_ABOVE	
4	DECIMAL	135	MNO_NOSTG_DSA	
4	DECIMAL	136	MNO_DSA_SIZE	
Component id.				
2	CHARACTER	SM	COMPID	
SM domain states.				
4	DECIMAL	1	PRE_INITIALISING	
4	DECIMAL	2	PRE_INITIALISED	
4	DECIMAL	3	INITIALISING	
4	DECIMAL	4	INITIALISED	
4	DECIMAL	5	QUIESCING	
4	DECIMAL	6	QUIESCED	
4	DECIMAL	7	TERMINATED	
Constants for Statistics				
4	DECIMAL	8192	STATS_BUFFER_SIZE	8K buffer
Pagesize.				
4	DECIMAL	4096	PAGESIZE	
4	HEX	FFFFF000	PAGEROUND	
The minimum fixed length value must be the size of QPF.				
4	DECIMAL	8	MIN_FIXED_LENGTH	

Len	Type	Value	Name	Description
Sizes of quickcell blocks.				
4	DECIMAL	4096	CTNBLOCK_SIZE	size of CTN block
4	DECIMAL	4096	DXEBLOCK_SIZE	size of DXE block
4	DECIMAL	4096	SATBLOCK_SIZE	size of SAT block
4	DECIMAL	4096	SCABLOCK_SIZE	size of SCA block
4	DECIMAL	4096	SCQBLOCK_SIZE	size of SCQ block
4	DECIMAL	4096	SMXBLOCK_SIZE	size of SMX block
4	DECIMAL	4096	SQEBLOCK_SIZE	size of SQE block
4	DECIMAL	4096	SUABLOCK_SIZE	size of SUA block
Index values for DSAs (used for indexing arrays in SMA and CAT). Note that these must be consistent with the values used for the DSA_NAME parameter in the various domain call parameter lists.				
4	DECIMAL	1	CDSA	
4	DECIMAL	2	UDSA	
4	DECIMAL	3	SDSA	
4	DECIMAL	4	RDSA	
4	DECIMAL	5	ECDSA	
4	DECIMAL	6	EUDSA	
4	DECIMAL	7	ESDSA	
4	DECIMAL	8	ERDSA	
4	DECIMAL	8	MAXDSA	
DSA names.				
8	CHARACTER	CDSA	CDSA_NAME	
8	CHARACTER	UDSA	UDSA_NAME	
8	CHARACTER	SDSA	SDSA_NAME	
8	CHARACTER	RDSA	RDSA_NAME	
8	CHARACTER	ECDSA	ECDSA_NAME	
8	CHARACTER	EUDSA	EUDSA_NAME	
8	CHARACTER	ESDSA	ESDSA_NAME	
8	CHARACTER	ERDSA	ERDSA_NAME	
Access values.				
4	DECIMAL	0	ACCESS_INVALID	
4	DECIMAL	1	ACCESS_CICS	
4	DECIMAL	2	ACCESS_USER	
4	DECIMAL	3	ACCESS_READ_ONLY	
Constants for self-tuning initial-free areas.				
4	DECIMAL	600	TUNING_INTERVAL	10 minutes
4	DECIMAL	604800	WEIGHTED_ AVERAGE_PERIOD	1 week
4	DECIMAL	1008	MAX_TUNING_ INTERVALS	
4	DECIMAL	4096	MIN_PRIMARY_SIZE	
4	DECIMAL	8192	MIN_SECONDARY_SIZE	
4	DECIMAL	65536	MAX_SECONDARY_ BELOW	
4	DECIMAL	1048576	MAX_SECONDARY_ ABOVE	
Maxpool is the upper bound of the array of pool chains whose dimension is 0:maxpool. Maxpool is calculated as (2 to the power n)-1 where n is the number of open TCB types that can inherit a subspace. n is defined by the dispatcher as num_ subspace_ open_types. (2 to the power of n) is defined by the dispatcher as combo_ subspace_ open_types.				
4	DECIMAL	1	MAXPOOL	
Total number of types of open TCB.				
1	DECIMAL	3	NUM_OPEN_TYPES	
Number of types of open TCB which can inherit subspaces (ie DSIT_ INHERIT_YES).				
1	DECIMAL	1	NUM_SUBSPACE_ OPEN_TYPES	
Number of combinations of types of open TCB which can inherit subspaces (ie DSIT_ INHERIT_ YES). This number is 2 to the power NUM_ SUBSPACE_ OPEN_TYPES.				
4	DECIMAL	2	COMBO_SUBSPACE_ OPEN_TYPES	

SMMCC Sm macro-compatability anchor block

SM domain Macro Compatibility Anchor block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	168	MCA	
(0)	CHARACTER	16	MCA_PREFIX	
(0)	UNSIGNED	2	MCA_LENGTH	
(2)	CHARACTER	1	MCA_ARROW	
(3)	CHARACTER	3	MCA_DFH	
(6)	CHARACTER	2	MCA_DOMID	
(8)	CHARACTER	8	MCA_BLOCK_NAME	
(10)	CHARACTER	8	*	reserved
(18)	CHARACTER	96	MCA_SUBPOOLS	macro subpool tokens/ids

SMSHRC24 subpool (SHARED_CIC24).

Offset Hex	Type	Len	Name (Dim)	Description
(18)	CHARACTER	12	*	
(18)	CHARACTER	8	MCA_SHRC24_ SPTOKEN	
(18)	ADDRESS	4	MCA_SHRC24_SPTOKEN_P	
(1C)	FULLWORD	4	*	
(20)	UNSIGNED	1	MCA_SHRC24_ SPID	
(21)	CHARACTER	3	*	

SMSHRU24 subpool (SHARED_ USER24).

Offset Hex	Type	Len	Name (Dim)	Description
(24)	CHARACTER	12	*	
(24)	CHARACTER	8	MCA_SHRU24_ SPTOKEN	
(24)	ADDRESS	4	MCA_SHRU24_SPTOKEN_P	
(28)	FULLWORD	4	*	
(2C)	UNSIGNED	1	MCA_SHRU24_ SPID	
(2D)	CHARACTER	3	*	

SMSHRC31 subpool (SHARED_ CIC31).

Offset Hex	Type	Len	Name (Dim)	Description
(30)	CHARACTER	12	*	
(30)	CHARACTER	8	MCA_SHRC31_ SPTOKEN	
(30)	ADDRESS	4	MCA_SHRC31_SPTOKEN_P	
(34)	FULLWORD	4	*	
(38)	UNSIGNED	1	MCA_SHRC31_ SPID	
(39)	CHARACTER	3	*	

SMSHRU31 subpool (SHARED_ USER31).

Offset Hex	Type	Len	Name (Dim)	Description
(3C)	CHARACTER	12	*	
(3C)	CHARACTER	8	MCA_SHRU31_ SPTOKEN	
(3C)	ADDRESS	4	MCA_SHRU31_SPTOKEN_P	
(40)	FULLWORD	4	*	
(44)	UNSIGNED	1	MCA_SHRU31_ SPID	
(45)	CHARACTER	3	*	

SMSHARED subpool (SHARED_ CIC24_SAA).

Offset Hex	Type	Len	Name (Dim)	Description
(48)	CHARACTER	12	*	
(48)	CHARACTER	8	MCA_SHARED_SPTOKEN	
(48)	ADDRESS	4	MCA_SHARED_SPTOKEN_P	
(4C)	FULLWORD	4	*	
(50)	UNSIGNED	1	MCA_SHARED_ SPID	
(51)	CHARACTER	3	*	

SMCONTROL subpool.

Offset Hex	Type	Len	Name (Dim)	Description
(54)	CHARACTER	12	*	
(54)	CHARACTER	8	MCA_CONTROL_SPTOKEN	
(54)	ADDRESS	4	MCA_CONTROL_SPTOKEN_P	
(58)	FULLWORD	4	*	
(5C)	UNSIGNED	1	MCA_CONTROL_ SPID	
(5D)	CHARACTER	3	*	

SMTP24 subpool.

Offset Hex	Type	Len	Name (Dim)	Description
(60)	CHARACTER	12	*	
(60)	CHARACTER	8	MCA_TP24_ SPTOKEN	
(60)	ADDRESS	4	MCA_TP24_SPTOKEN_P	
(64)	FULLWORD	4	*	
(68)	UNSIGNED	1	MCA_TP24_SPID	
(69)	CHARACTER	3	*	

SMTP subpool.

Offset Hex	Type	Len	Name (Dim)	Description
(6C)	CHARACTER	12	*	
(6C)	CHARACTER	8	MCA_TP_SPTOKEN	
(6C)	ADDRESS	4	MCA_TP_ SPTOKEN_P	
(70)	FULLWORD	4	*	
(74)	UNSIGNED	1	MCA_TP_SPID	
(75)	CHARACTER	3	*	
Flags.				
(78)	CHARACTER	4	*	
(78)	BITSTRING	1	*	
	1...		MCA_SMMC_ ACTIVE	INITIALISE function completed
	.111 1111		*	reserved
(79)	BITSTRING	3	*	reserved
(7C)	FULLWORD	4	*	reserved
(80)	FULLWORD	4	*	reserved
(84)	FULLWORD	4	*	reserved
(88)	FULLWORD	4	*	reserved
(8C)	FULLWORD	4	*	reserved
(90)	FULLWORD	4	*	reserved
(94)	FULLWORD	4	*	reserved
(98)	FULLWORD	4	*	reserved
(9C)	FULLWORD	4	*	reserved
(A0)	FULLWORD	4	*	reserved
(A4)	FULLWORD	4	*	reserved
(A8)	CHARACTER		*	

SHARED/CONTROL subpool SAA.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	SHR	
(0)	CHARACTER	4	SHR_SAA	
(0)	CHARACTER	1	SHR_CLASS	
(1)	CHARACTER	1	SHR_INITIMG	
(2)	UNSIGNED	2	SHR_LENGTH	
(4)	CHARACTER	*	SHR_DATA	

User storage SAA.
Note that the address field points to the TCA.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	USR	
(0)	CHARACTER	8	USR_SAA	
(0)	CHARACTER	1	USR_CLASS	
(1)	CHARACTER	1	USR_INITIMG	
(2)	UNSIGNED	2	USR_LENGTH	
(4)	ADDRESS	4	USR_TCAP	
(8)	CHARACTER	*	USR_DATA	

TP storage SAA.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	13	TPE	
(0)	CHARACTER	8	TPE_SAA	
(0)	CHARACTER	1	TPE_CLASS	
(1)	CHARACTER	1	TPE_INITIMG	
(2)	UNSIGNED	2	TPE_LENGTH	
(4)	ADDRESS	4	TPE_NEXT	
(8)	CHARACTER		TPE_LIOA_ DATA_START	
(8)	CHARACTER	5	TPE_TIOA_PREFIX	
(D)	CHARACTER		TPE_TIOA_ DATA_START	

Constants

Len	Type	Value	Name	Description
8	CHARACTER	SMSHARED	SPNAME_SHARED	
8	CHARACTER	SMSHRC24	SPNAME_SHRC24	
8	CHARACTER	SMSHRU24	SPNAME_SHRU24	
8	CHARACTER	SMSHRC31	SPNAME_SHRC31	
8	CHARACTER	SMSHRU31	SPNAME_SHRU31	
8	CHARACTER	SMCONTRL	SPNAME_CONTROL	
8	CHARACTER	SMTP24	SPNAME_TP24	
8	CHARACTER	SMTP	SPNAME_TP	
Miscellaneous constants.				
4	DECIMAL	65520	MAX_SHARED_CICS24_SAA_LENGTH	
4	DECIMAL	65515	MAX_TIOA_LENGTH	
4	DECIMAL	65520	MAX_LIOA_LENGTH	
4	DECIMAL	65520	MAX_CICS24_ SAA_LENGTH	
1	HEX	80	GETFLAG	
1	HEX	7F	GETFLAG_OFF	
Following is used by storage recovery when an SAA has been found to be invalid.				
1	DECIMAL	0	INVALID_CLASS	
1	HEX	0A	TCACLASS	

SOA Sockets anchor block

-

This anchor block contains the global storage for the SO domain.

It defines the domain state information, variables and constants required by the SO gates and other external programs such as DFHSOTRI, the domain trace interpretation routine.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	328	SOA	
(0)	CHARACTER	16	SOA_PREFIX	eyecatcher
(0)	HALFWORD	2	SOA_LENGTH	total length of soa
(2)	CHARACTER	1	SOA_ARROW	>
(3)	CHARACTER	3	SOA_DFH	DFH
(6)	CHARACTER	2	SOA_DOMID	SO
(8)	CHARACTER	8	SOA_BLOCK_NAME	ANCHOR
(10)	UNSIGNED	1	SOA_SO_STATE	SO domain state initialised, quiesced or terminated
(11)	UNSIGNED	1	SOA_LISTENER_ STATE	SO Listener state
(12)	CHARACTER	1	SOA_FLAGS1	
	1...		SOA_TCPIP_ REQUIRED	TCPIP support requested
	.1..		SOA_COLD_START	CICS cold started
	..1.		SOA_SELECT_WAIT	Listener in select
	...1 1111		*	
(13)	CHARACTER	1	SOA_FLAGS2	SSL flags
	1...		SOA_SSL_ AVAILABLE	SSL available
	.111 11..		*	Reserved
	.... ..1.		SOA_STRONG_ ENCRYPTION	ENCRYPTION=STRONG
	1		SOA_NORMAL_ ENCRYPTION	ENCRYPTION=NORMAL
(14)	CHARACTER	1	SOA_FLAGS3	
	1...		SOA_XRSINDI_ ACTIVE	XRSINDI exit active
	.1..		SOA_NAMESERVER_ ERR	Nameservice unavailable
	..11 1111		*	
(15)	UNSIGNED	1	*	Reserved
(16)	UNSIGNED	1	*	Reserved
(17)	UNSIGNED	1	*	Reserved
(18)	FIXED	4	SOA_SELECTEX_ECB	ECB for selectex
(18)	UNSIGNED	1	POST_BYTE	
(19)	UNSIGNED	3	COMPLETION_CODE	
(1C)	ADDRESS	4	SOA_LOCK_TOKEN	SO domain lock token

Offset Hex	Type	Len	Name (Dim)	Description
(20)	ADDRESS	4	SOA_SO_MODENAME_TOKEN	SO TCB Modename token
(24)	ADDRESS	4	SOA_TCPIPSERVICE_LOCK_TOKEN	
(28)	ADDRESS	4	SOA_TCBPOOL_LOCK_TOKEN	S8 TCB pool lock
(2C)	ADDRESS	4	SOA_SL_MODENAME_TOKEN	SL TCB Modename token
(30)	ADDRESS	4	SOA_SOIS_CEEPIPI_TOKEN	CEEPIPI token
(34)	STRUCTURE IsA(ETOKEN)	8	SOA_SO_STOKEN	Subspace Token
(34)	ADDRESS	4	P	
(38)	FULLWORD	4	N	
(3C)	STRUCTURE IsA(ETOKEN)	8	SOA_GENERAL_ SPTOKEN	SOGENRL subpool token
(3C)	ADDRESS	4	P	
(40)	FULLWORD	4	N	
(44)	STRUCTURE IsA(ETOKEN)	8	SOA_LTE_SPTOKEN	SOLTE subpool token
(44)	ADDRESS	4	P	
(48)	FULLWORD	4	N	
(4C)	STRUCTURE IsA(ETOKEN)	8	SOA_STE_SPTOKEN	SOSTE subpool token
(4C)	ADDRESS	4	P	
(50)	FULLWORD	4	N	
(54)	STRUCTURE IsA(ETOKEN)	8	SOA_SO_TCB_TOKEN	TCB token for SOCKETS
(54)	ADDRESS	4	P	
(58)	FULLWORD	4	N	
(5C)	STRUCTURE IsA(ETOKEN)	8	SOA_SL_TCB_TOKEN	TCB token for LISTENER
(5C)	ADDRESS	4	P	
(60)	FULLWORD	4	N	
(64)	ADDRESS	4	SOA_DFHSOSE_ ENTRY	Address of DFHSOSE
(68)	ADDRESS	4	SOA_CEEPIPI_ ENTRY	Address of CEEPIPI
(70)	CHARACTER	36	SOA_LTE_CHAIN	
(70)	FULLWORD	4	SOA_LTE_ NUM_ENTRIES	Number of LTEs
(74)	FIXED	4	SOA_LTE_ EMPTY_ECB	Posted when empty
(74)	UNSIGNED	1	POST_BYTE	
(75)	UNSIGNED	3	COMPLETION_ CODE	
(78)	CHARACTER	28	SOA_LTE_HEAD	LTE chain header block
(98)	CHARACTER	80	SOA_GSK	GSK interface data
(98)	CHARACTER	48	SOA_KEYFILE_PATHNAME	Keyring file path
(C8)	CHARACTER	16	SOA_KEYFILE_PASSWORD	Keyfile password
(D8)	FULLWORD	4	SOA_SSLV2_ TIMEOUT	V2 timeout (secs)
(DC)	FULLWORD	4	SOA_SSLV3_ TIMEOUT	V3 timeout (secs)
(E0)	CHARACTER	1	SOA_DFHSOSE_ SUFFIX	Security suffix
(E1)	CHARACTER	3	*	Reserved
(E4)	ADDRESS	4	SOA_SSL_ SUBTASKS	SSL subtask block
(E8)	ADDRESS	4	SOA_TCPIPSERVICE_CLASSP	tcpipservice chain
(EC)	UNSIGNED	4	SOA_TOKEN_ COUNTER	Counter for unique toks
(F0)	CHARACTER	8	SOA_LAST_ RESET_TIME	Time (STCK) that global stats were last reset
(F8)	ADDRESS	4	SOA_STATS_ BUFFER_PTR	Stats return buffer
(FC)	CHARACTER	76	SOA_WLM_DATA	
(FC)	UNSIGNED	1	SOA_WLM_STATE	DDNS availabliliy
(FD)	CHARACTER	3	*	Reserved
(100)	CHARACTER	8	SOA_WLM_ SERVERNAME	Server name (APPLID)
(108)	CHARACTER	64	SOA_WLM_ HOSTNAME	Host Name
(148)	CHARACTER		*	Alignment

-

There is one LTE for each listening socket that is handled by the SO domain listener. The lte_port is kept in the prefix for sorting and searching. The chain of LTE's is kept sorted in ascending order of port number.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	624	LTE	
(0)	CHARACTER	28	LTE_PREFIX	
(0)	HALFWORD	2	LTE_LENGTH	total length of lte
(2)	CHARACTER	1	LTE_ARROW	>
(3)	CHARACTER	3	LTE_DFH	DFH
(6)	CHARACTER	2	LTE_DOMID	SO
(8)	CHARACTER	8	LTE_BLOCK_NAME	LTE
(10)	ADDRESS	4	LTE_NEXT	-> next LTE (or header)
(14)	ADDRESS	4	LTE_PREV	-> prev LTE (or header)
(18)	UNSIGNED	2	LTE_PORT	Port number
(1A)	UNSIGNED	2	*	Reserved for alignment
(1C)	CHARACTER	36	LTE_STE_CHAIN	STE chain from this LTE
(1C)	FIXED	4	LTE_STE_ EMPTY_ECB	ECB posted when empty
(1C)	UNSIGNED	1	POST_BYTE	
(1D)	UNSIGNED	3	COMPLETION_ CODE	
(20)	FULLWORD	4	LTE_STE_ NUM_ENTRIES	# STE's in chain
(24)	CHARACTER	28	LTE_STE_HEAD	STE chain header block
(40)	UNSIGNED	4	LTE_CONNECTION_ COUNT	Current no of open sockets
(44)	UNSIGNED	4	LTE_IDENTITY_NO	Unique number for identity
(48)	FIXED	4	LTE_READY_ECB	ECB for LTE ready
(48)	UNSIGNED	1	POST_BYTE	
(49)	UNSIGNED	3	COMPLETION_CODE	
(4C)	UNSIGNED	4	LTE_LISTEN_ BACKLOG	Backlog value for listen
(50)	FULLWORD	4	LTE_SOCKET	Socket descriptor
(54)	CHARACTER	1	LTE_FLAG1	
	1...		LTE_NEW	Newly created by register
	.1..		LTE_SOCKET_ CREATED	BPX1SOC called
	..1.		LTE_SOCKET_ BOUND	BPX1LTN called
	...1		LTE_SOCKET_ LISTENED	BPX1BND called
	.... 1...		LTE_SOCKET_ GETCLID	BPX1CLD called
	.... .1..		LTE_SOCKET_ CLOSED	BPX1CLO called
	.... ..1.		LTE_DEREGISTERING	Processing deregister
	1		LTE_IMMCLOSING	Processing immclose
(55)	CHARACTER	1	LTE_FLAG2	
	1...		LTE_CONNECTION_ FAILURE	A connection has failed
	.111 1111		*	Reserved
(58)	CHARACTER	276	LTE_SERVER_ ADDRESS_AREA	Server address area
(58)	CHARACTER	256	LTE_SERVER_ HOSTNAME_BUF	Hostname buffer
(158)	UNSIGNED	1	LTE_SERVER_ HOSTNAME_LEN	Length of hostname
(159)	CHARACTER	15	LTE_SERVER_ IP_ADDRESS	IP address string
(168)	UNSIGNED	4	LTE_SERVER_ BIN_IP_ADDR	Binary address
(16C)	CHARACTER	33	LTE_SERVICE_AREA	
(16C)	CHARACTER	8	LTE_SERVICE_ NAME	Name of service eg. HTTP
(174)	CHARACTER	8	LTE_SERVICE_URM	Name of URM for service
(17C)	CHARACTER	4	LTE_SERVICE_ TRANID	Transaction to attach
(180)	CHARACTER	6	LTE_SERVICE_ TSQPREFIX	TSQ Prefix
(186)	CHARACTER	2	*	Reserved
(188)	FULLWORD	4	LTE_RECV_ TIMEOUT	Receive timeout value
(18C)	UNSIGNED	1	LTE_SERVICE_ FLAGS	
	1...		LTE_SERVICE_ SSL	Secure Sockets Layer
	.1..		LTE_SERVICE_ CLIAUTH	Client authentication
(190)	CHARACTER	30	LTE_WLM_DATA	Work Load Manager
(190)	UNSIGNED	1	LTE_WLM_STATE	Reg/De-reg State

Offset Hex	Type	Len	Name (Dim)	Description
(191)	CHARACTER	1	LTE_WLM_FLAGS	Reserved
	1...		LTE_WLM_ CRITIAL	Group_Critical
	.111 1111		*	Reserved
(192)	CHARACTER	2	*	Reserved
(194)	UNSIGNED	4	LTE_WLM_RETCODE	Last Return code
(198)	UNSIGNED	4	LTE_WLM_RSNCODE	Last Reason code
(19C)	CHARACTER	18	LTE_WLM_ GROUPNAME	Group name
(1AE)	CHARACTER	110	LTE_SOCKADDR	
(1AE)	STRUCTURE IsA(SOCK_HEADER)	2	LTE_SOCKADDR_ HEADER	
				SockAddr
(1AE)	UNSIGNED	1	SOCK_LEN	Address length
(1AF)	UNSIGNED	1	SOCK_FAMILY	Address family
(1B0)	CHARACTER		SOCK_DATA	Protocol specific area
(1B0)	CHARACTER	108	LTE_ADDR	structure for
(1B0)	STRUCTURE IsA(SOCK_INET_PART)	14	LTE_INET_ADDR	the host
(1B0)	UNSIGNED	2	SOCK_SIN_PORT	Port number used by the appl
(1B2)	CHARACTER	4	SOCK_SIN_ADDR	Inet addr (netid)
(1B6)	CHARACTER	8	*	unused
(1B0)	STRUCTURE IsA(SOCK_UNIX_PART)	108	LTE_UNIX_ADDR	machine.
Deleted field use SOCK_LEN instead Length of the path name				
(1B0)	CHARACTER	108	SOCK_SUN_NAME	Path name of the socket
(220)	CHARACTER	40	LTE_STATISTICS_ DATA	
				Statistics collection data
(220)	CHARACTER	8	LTE_SEND_BYTES	Bytes sent 64 bits
(220)	BITSTRING	4	LTE_SEND_ BYTES_HIGH	
				* Need to split into
(224)	BITSTRING	4	LTE_SEND_ BYTES_LOW	
				* 32 bit values for C
(228)	CHARACTER	8	LTE_RECV_BYTES	Bytes received
(228)	BITSTRING	4	LTE_RECV_ BYTES_HIGH	
				* Need to split into
(22C)	BITSTRING	4	LTE_RECV_ BYTES_LOW	
				* 32 bit values for C
(230)	CHARACTER	8	LTE_OPEN_TIME	Open time (STCK)
(230)	BITSTRING	4	LTE_OPEN_ TIME_HIGH	
(234)	BITSTRING	4	LTE_OPEN_ TIME_LOW	
(238)	FULLWORD	4	LTE_SEND_COUNT	# of sends
(23C)	FULLWORD	4	LTE_RECV_COUNT	# if receives
(240)	FULLWORD	4	LTE_ATTACH_ COUNT	# service attaches
(244)	FULLWORD	4	LTE_PEAK_CONN	highest # connections
(248)	CHARACTER	40	LTE_CID	
(270)	CHARACTER		*	

--
-

There is one STE for each socket that is created using accept.
These represent the individual sessions to clients. The
soa_ ste_head contains 0 for the ste_ prev pointer.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	544	STE	
(0)	CHARACTER	28	STE_PREFIX	
(0)	HALFWORD	2	STE_LENGTH	total length of ste
(2)	CHARACTER	1	STE_ARROW	>
(3)	CHARACTER	3	STE_DFH	DFH
(6)	CHARACTER	2	STE_DOMID	SO
(8)	CHARACTER	8	STE_BLOCK_NAME	STE
(10)	ADDRESS	4	STE_NEXT	-> next STE (or header)
(14)	ADDRESS	4	STE_PREV	-> prev STE (or header)
(18)	FULLWORD	4	STE_SOCKET	Socket descriptor
(1C)	CHARACTER	1	STE_FLAG1	
	1...		STE_SOCKET_ GIVEN	GIVESOCKET called
	.1..		STE_SOCKET_ TAKEN	TAKESOCKET called
	..1.		STE_SOCKET_ CLOSED	Socket has closed
	...1		STE_SESSION_ ERROR	Error occured
	 1111		*	Reserved
(1D)	CHARACTER	1	STE_FLAG2	SSL flags
	1...		STE_SSL_ REQUIRED	Secure Sockets Layer
	.1..		STE_SSL_ COMPLETE	SSL handshake done
	..11 1111		*	Reserved
(1E)	UNSIGNED	2	*	Reserved
(20)	CHARACTER	4	STE_CS_FLAG_WORD	Flag word for compare&swap
(20)	CHARACTER	1	STE_CS_ FLAG_BYTE1	
	1...		STE_TERMINATION	Set for termination

Offset Hex	Type	Len	Name (Dim)	Description
	.1..		STE_SEND_ IN_PROGRESS	Set for a send
	..1.		STE_RECV_ IN_PROGRESS	Set for a recv
	...1 1111		*	Reserved
(21)	BITSTRING	3	*	Reserved
(24)	UNSIGNED	4	STE_TXN_COUNT	# of referencing txns
(28)	UNSIGNED	4	STE_REF_COUNT	# of pointers to STE
(2C)	CHARACTER	8	STE_SERVICE_ LTE_TOKEN	Originating LTE
(2C)	ADDRESS	4	STE_SERVICE_ LTE_PTR	Pointer to LTE
(30)	UNSIGNED	4	STE_SERVICE_ LTE_ID	Identity number of LTE
(34)	UNSIGNED	4	STE_IDENTITY_NO	Unique number for identity
(38)	FULLWORD	4	STE_RECV_TIMEOUT	Timeout for receives
(3C)	FIXED	4	STE_RECV_ ASYNC_ECB	ECB for recv waits
(3C)	UNSIGNED	1	POST_BYTE	
(3D)	UNSIGNED	3	COMPLETION_CODE	
(40)	FIXED	4	STE_SEND_ ASYNC_ECB	ECB for send waits
(40)	UNSIGNED	1	POST_BYTE	
(41)	UNSIGNED	3	COMPLETION_CODE	
(44)	FULLWORD	4	STE_SSL_HANDLE	Secure socket handle
(48)	ADDRESS	4	STE_SSL_ THREAD_PTR	SSL thread descriptor
(4C)	ADDRESS	4	STE_SSL_LE_TOKEN	SSL LE environment
(50)	CHARACTER	24	STE_ERROR_DATA	
(50)	BITSTRING	2	STE_ERROR_CODE	Trace point of error
(52)	UNSIGNED	1	*	Reserved
(53)	UNSIGNED	1	STE_ERROR_ FUNCTION	CDURUN function number@P3A
(54)	FULLWORD	4	STE_ERROR_ FORMAT	CDURUN format number
(58)	CHARACTER	4	STE_ERROR_ TRANID	Tranid of error task
(5C)	CHARACTER	4	STE_ERROR_ TRANNUM	Trannum of error task
(60)	UNSIGNED	4	STE_ERROR_ RESPONSE	Response for error
(64)	UNSIGNED	4	STE_ERROR_ REASON	Reason for error
(68)	CHARACTER	15	STE_CLIENT_ IP_ADDRESS	IP address
(77)	CHARACTER	9	STE_USERID_ STRUCTURE	
(77)	UNSIGNED	1	STE_USERID_LEN	Length of userid
(78)	CHARACTER	8	STE_USERID	Certificate userid
(80)	STRUCTURE IsA(ETOKEN)	8	STE_REPOSITORY_ TOKEN	Repository token
(80)	ADDRESS	4	P	
(84)	FULLWORD	4	N	
(88)	CHARACTER	110	STE_SOCKADDR	Client sockaddr
(88)	STRUCTURE IsA(SOCK_HEADER)	2	STE_SOCKADDR_ HEADER	
(88)	UNSIGNED	1	SOCK_LEN	Address length
(89)	UNSIGNED	1	SOCK_FAMILY	Address family
(8A)	CHARACTER		SOCK_DATA	Protocol specific area
(8A)	CHARACTER	108	STE_ADDR	
(8A)	STRUCTURE IsA(SOCK_INET_PART)	14	STE_INET_ADDR	
(8A)	UNSIGNED	2	SOCK_SIN_PORT	Port number used by the appl
(8C)	CHARACTER	4	SOCK_SIN_ADDR	Inet addr (netid)
(90)	CHARACTER	8	*	unused
(8A)	STRUCTURE IsA(SOCK_UNIX_PART)	108	STE_UNIX_ADDR	
(8A)	CHARACTER	108	SOCK_SUN_NAME	Path name of the socket
(F8)	CHARACTER	128	STE_SAIOCB	Async send CB
(178)	CHARACTER	128	STE_RAIOCB	Async recv CB
(1F8)	CHARACTER	40	STE_CID	ClientID
(220)	CHARACTER		*	

--
-

These structures represents a pool of TCBs that are set aside for Secure Sockets Layer.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	344	SSL_SUBTASK_ VECTOR	
(0)	CHARACTER	16	SSLT_PREFIX	
(0)	HALFWORD	2	SSLT_LENGTH	Total length of SSLTCBV

Offset Hex	Type	Len	Name (Dim)	Description
(2)	CHARACTER	1	SSLT_ARROW	>
(3)	CHARACTER	3	SSLT_DFH	DFH
(6)	CHARACTER	2	SSLT_DOMID	SO
(8)	CHARACTER	8	SSLT_BLOCK_NAME	SSLTCBV
(10)	UNSIGNED	4	SSLT_TCB_ COUNTERS	Fullword container
(10)	HALFWORD	2	SSLT_MAX_TCBS	Total TCB entries
(12)	HALFWORD	2	SSLT_ACTIVE_ TCBS	TCB entries in use
(14)	UNSIGNED	4	SSLT_MODE_TOKEN	Mode token
(18)	CHARACTER	32	SSLT_TCB_ENTRY (0 9)	Allocate 10 TCB slots

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	SSLTCB_ENTRY	
(0)	BITSTRING	1	SSLT_FLAG1	First flag byte
	1...		SSLT_BUSY	SSLT entry in use
	.111		*	Reserved
	.... 1...		SSLT_INITIALIZED	Thread initialized
	.... .1..		*	Reserved
	.... ..1.		SSLT_INIT_ STARTED	Init in progress
	1		SSLT_INIT_ FAILED	Initialization failed
(1)	BITSTRING	1	SSLT_FLAG2	Second flag byte
(2)	CHARACTER	2	*	Reserved
(4)	ADDRESS	4	*	Reserved
(8)	ADDRESS	4	SSLT_STE_ADDRESS	Associated STE address
(C)	ADDRESS	4	SSLT_TCB_ADDRESS	Associated TCB address
(10)	STRUCTURE IsA(ETOKEN)	8	SSLT_TCB_TOKEN	Dispatcher's TCB token
(10)	ADDRESS	4	P	
(14)	FULLWORD	4	N	
(18)	UNSIGNED	4	SSLT_CEEPIPI_ TOKEN	LE environment token
(1C)	UNSIGNED	4	*	Reserved
(20)	CHARACTER		*	

--
-

This structure holds all the parameter information and related
data for the OpenEdition Assembler Callable Service (BPX) calls.
It is heavily for tracing information.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	64	BPX_INTERFACE	
(0)	FULLWORD	4	BPX_RETURN_VALUE	
(4)	FULLWORD	4	BPX_RETURN_CODE	
(8)	FULLWORD	4	BPX_REASON_CODE	
(C)	CHARACTER	4	*	
(10)	ADDRESS	4	BPX_STE_PTR	
(14)	ADDRESS	4	BPX_LTE_PTR	
(18)	CHARACTER	40	BPX_PARAMETERS	
(18)	CHARACTER	8	ASYNCIO_PARMS	
(18)	UNSIGNED	4	AIOCB_LEN	
(1C)	ADDRESS	4	AIOCB_ADDR	
(18)	CHARACTER	40	SELECT_PARMS	
(18)	UNSIGNED	4	NUMBER_MSGSFDS	
(1C)	UNSIGNED	4	READ_LIST_ LENGTH	
(20)	ADDRESS	4	READ_LIST_ADDR	
(24)	UNSIGNED	4	WRITE_LIST_ LENGTH	
(28)	ADDRESS	4	WRITE_LIST_ ADDR	
(2C)	UNSIGNED	4	EXCEPTION_ LIST_LENGTH	
(30)	ADDRESS	4	EXCEPTION_ LIST_ADDR	
(34)	ADDRESS	4	TIMEOUT_ POINTER	
(38)	ADDRESS	4	ECB_POINTER	
(3C)	UNSIGNED	4	USER_OPTION_ FIELD	
(18)	CHARACTER	20	SOCKET_PARMS	
(18)	UNSIGNED	4	DOMAIN	
(1C)	UNSIGNED	4	TYPE	
(20)	UNSIGNED	4	PROTOCOL	
(24)	UNSIGNED	4	DIMENSION	
(28)	UNSIGNED	4	SOCKET_VECTOR	
(18)	CHARACTER	12	BIND_PARMS	
(18)	UNSIGNED	4	SOCKET_ DESCRIPTOR	
(1C)	UNSIGNED	4	SOCKADDR_ LENGTH	
(20)	ADDRESS	4	SOCKADDR_ADDR	
(18)	CHARACTER	8	LISTEN_PARMS	
(18)	UNSIGNED	4	SOCKET_ DESCRIPTOR	
(1C)	UNSIGNED	4	BACKLOG	

Offset Hex	Type	Len	Name (Dim)	Description
(18)	CHARACTER	12	ACCEPT_PARMS	
(18)	UNSIGNED	4	SOCKET_ DESCRIPTOR	
(1C)	UNSIGNED	4	SOCKADDR_ LENGTH	
(20)	ADDRESS	4	SOCKADDR_ADDR	
(18)	CHARACTER	16	GETCLIENTID_ PARMS	
(18)	UNSIGNED	4	FUNCTIONCODE	
(1C)	UNSIGNED	4	DOMAIN	
(20)	UNSIGNED	4	CLIENTID_ LENGTH	
(24)	ADDRESS	4	CLIENTID_ADDR	
(18)	CHARACTER	12	GETHOSTNAME_ PARMS	
(18)	UNSIGNED	4	DOMAIN	
(1C)	UNSIGNED	4	NAME_LENGTH	
(20)	ADDRESS	4	NAME_ADDR	
(18)	CHARACTER	12	TAKESOCKET_ PARMS	
(18)	UNSIGNED	4	CLIENTID_ LENGTH	
(1C)	ADDRESS	4	CLIENTID_ADDR	
(20)	UNSIGNED	4	SOCKET_ID	
(18)	CHARACTER	12	GIVESOCKET_ PARMS	
(18)	UNSIGNED	4	SOCKET_ DESCRIPTOR	
(1C)	UNSIGNED	4	CLIENTID_ LENGTH	
(20)	ADDRESS	4	CLIENTID_ADDR	
(18)	CHARACTER	4	CLOSE_PARMS	
(18)	UNSIGNED	4	FILE_DESCRIPTOR	
(18)	CHARACTER	24	SETSOCKOPT_ PARMS	
(18)	UNSIGNED	4	SOCKET_ DESCRIPTOR	
(1C)	UNSIGNED	4	OPERATION	
(20)	UNSIGNED	4	LEVEL	
(24)	UNSIGNED	4	OPTION_NAME	
(28)	UNSIGNED	4	OPTION_ DATA_LENGTH	
(2C)	ADDRESS	4	OPTION_ DATA_ADDR	
(18)	CHARACTER	20	SIGPROCMASK_ PARMS	
(18)	UNSIGNED	4	HOW	
(1C)	CHARACTER	8	NEW_SIGNAL_ MASK	
(24)	CHARACTER	8	OLD_SIGNAL_ MASK	

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	SO_STATE_ INITIALISING	
1	DECIMAL	2	SO_STATE_ INITIALISED	
1	DECIMAL	3	SO_STATE_QUIESCING	
1	DECIMAL	4	SO_STATE_QUIESCED	
1	DECIMAL	5	SO_STATE_ TERMINATED	
1	DECIMAL	1	SO_LISTENER_STATE_OPEN	
1	DECIMAL	2	SO_LISTENER_STATE_OPENING	
1	DECIMAL	3	SO_LISTENER_STATE_CLOSED	
1	DECIMAL	4	SO_LISTENER_STATE_CLOSING	
1	DECIMAL	5	SO_LISTENER_STATE_IMMCLOSING	
1	DECIMAL	0	SO_SERVICE_WLM_STATE_NOTAPPLIC	
1	DECIMAL	1	SO_SERVICE_WLM_STATE_AVAILABLE	
1	DECIMAL	2	SO_SERVICE_WLM_STATE_UNAVAILABLE	
1	DECIMAL	3	SO_SERVICE_WLM_STATE_REGISTERED	
1	DECIMAL	4	SO_SERVICE_WLM_STATE_UNREGISTERED	
1	DECIMAL	5	SO_SERVICE_WLM_STATE_REGERROR	
1	DECIMAL	6	SO_SERVICE_WLM_STATE_DEREGISTERED	
1	DECIMAL	7	SO_SERVICE_WLM_STATE_DEREGERROR	

STAFB Statistics authorised parameter block

```
Segment Name = DFHSTAFB
DESCRIPTIVE NAME = CICS/MVS Statistics (ST) Domain
                Authorised Facilities Parameter Block
Function =
    This file contains the control block and constant
    declarations for the parameter list used by Statistics
    for communication between the functional gate and the
    SVC service routine.
Notes:
 Dependencies = S/370
 Restrictions = none
 Register Conventions = domain standard (no special usage)
 Patch Label = N/A
 Module Type = N/A
 Attributes = N/A
Statistics Authorised Facilities Parm Block -- S A F P B -
This contains:
        The authorised facility function code.
        The function return code.
        The SMF record address
        The creation time of the SAFPB
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	68	SAFPB	
(0)	CHARACTER	16	SAFPB_PREFIX	
(0)	HALFWORD	2	SAFPB_LENGTH	Length
(2)	CHARACTER	1	SAFPB_ARROW	Arrow
(3)	CHARACTER	3	SAFPB_DFH	DFH
(6)	CHARACTER	2	SAFPB_DOMAIN	ST
(8)	CHARACTER	8	SAFPB_BLOCK_ID	SAFPB
(10)	UNSIGNED	2	SAFPB_FUNCTION	Function SMFWTM
(12)	UNSIGNED	1	SAFPB_RESPONSE	Response
(13)	BITSTRING	1	*	
	1...		SAFPB_GTF_ TRACE_FLAG	GTF flag
	.111 1111		*	
(14)	ADDRESS	4	SAFPB_SMF_RECORD	-> SMF buffer
(18)	ADDRESS	4	*	Reserved
(1C)	UNSIGNED	1	SAFPB_SMF_RC	SMF response
(1D)	UNSIGNED	1	*	Reserved
(1E)	UNSIGNED	2	*	
(20)	FULLWORD	4	SAFPB_RTNREG0	MVS rtnreg 0
(24)	FULLWORD	4	SAFPB_RTNREG1	MVS rtnreg 1
(28)	FULLWORD	4	SAFPB_RTNREG15	MVS rtnreg 15
(2C)	UNSIGNED	4	*	Reserved
(30)	UNSIGNED	4	*	Reserved
(34)	CHARACTER	8	*	Reserved
(3C)	CHARACTER	8	SAFPB_CREATION_ STCK	Creation time
(44)	CHARACTER		*	

Constants

Len	Type	Value	Name	Description
2	DECIMAL	1	SAFPB_SMFEWTM	
0	BIT	1	SAFPB_GTF_TRACE_ON	
0	BIT	0	SAFPB_GTF_ TRACE_OFF	
1	DECIMAL	0	SAFPB_OK	
1	DECIMAL	1	SAFPB_NO_FESTAE	
1	DECIMAL	2	SAFPB_NO_ STORAGE_253	
1	DECIMAL	3	SAFPB_NO_ AUTHORISATION	
1	DECIMAL	4	SAFPB_NO_ STORAGE_SMF	
1	DECIMAL	5	SAFPB_INVALID_ RECORD_LENGTH	
1	DECIMAL	6	SAFPB_NOT_ CICS_RECORD	
1	DECIMAL	7	SAFPB_SMF_ERROR	
1	DECIMAL	254	SAFPB_INVALID_ FUNCTION	*

STCB1 Statistics domain anchor block

```
Segment Name = DFHSTCB1
DESCRIPTIVE NAME = CICS/MVS Statistics Domain (ST)
               Control Blocks 1.
Function =
    This file contains the data structure
    declarations used by the Statistics Domain.
    The data structure is :
        ANCHOR - ST Anchor block
        CATALOG_RECORD - ST CC Catalog record
        USS_BUFFER - Chain USS records
Notes:
 Dependencies = S/370
 Restrictions = none
 Register Conventions = domain standard (no special usage)
 Patch Label = N/A
 Module Type = N/A
 Attributes = N/A
                          RECORD_STATISTICS
                          trandefs
                         prolog to be generated
 ST anchor block
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	132	ANCHOR	Anchor Block
(0)	CHARACTER	16	ANC_PREFIX	Anchor prefix area
(0)	HALFWORD	2	ANC_LENGTH	Anchor length
(2)	CHARACTER	1	ANC_ARROW	Arrow eyecatcher
(3)	CHARACTER	3	ANC_DFH	DFH
(6)	CHARACTER	2	ANC_DOMID	Domain id
(8)	CHARACTER	8	ANC_BLOCK_NAME	Control block name
(10)	CHARACTER	43	COLLECTION_ MANAGEMENT	Collection management
(10)	CHARACTER	8	CM_INTERVAL	
(10)	UNSIGNED	4	CM_INT_SEC	Collection interval
(14)	UNSIGNED	4	CM_INT_ MICROSEC	
(18)	CHARACTER	8	CM_INTERVAL_ TOKEN	Token from Timer
(20)	CHARACTER	6	CM_END_ OF_DAY_TIME	EOD collection time
(26)	CHARACTER	8	CM_END_ OF_DAY_TOKEN	Token from Timer
(2E)	CHARACTER	6	CM_PEND_ RESET_TIME	Pending reset time hhmmss *
(34)	CHARACTER	6	CM_PREV_ RESET_TIME	Previous reset time hhmmss *
(3A)	BITSTRING	1	CM_FLAGS	Flags
	1...		CM_COLLECT_ OPTION	Collect option
	.1..		CM_USS_OPTION	USS stats?
	..1.		*	unused
	...1		*	unused
	.... 1...		*	unused
	.... .1..		*	unused
	.... ..1.		*	unused
	1		*	unused
(3B)	BITSTRING	1	ANC_FLAGS	Anchor flags
	1...		*	Reserved
	.1..		ANC_SYSTEM_ TERMINATING	set by terminating EOD collection
	..1.		ANC_USER_ EXIT_STATUS	user exit ON/OFF
	...1		*	unused
	.... 1...		*	unused
	.... .1..		*	unused
	.... ..1.		*	unused
	1		*	unused
(3C)	CHARACTER	3	*	filler
(3F)	UNSIGNED	1	LAST_SMF_RC	Last SMF ret. code received
(40)	CHARACTER	8	SUBPOOL_TOKEN	Obtained from SM
(48)	ADDRESS	4	LOCK_TOKEN	Obtained from LM
(4C)	ADDRESS	4	USS_LOCK_TOKEN	
(50)	ADDRESS	4	SMF_PTR	-> to SMF buffer
(54)	ADDRESS	4	SAFPB_PTR	-> to SAFPB
(58)	ADDRESS	4	STATISTICS_PTR	-> to ST Domain Stats Rec.
(5C)	ADDRESS	4	USS_CHAIN_PTR	USS record chain
(60)	UNSIGNED	1	DOMAIN_STATUS	Domain status - Initialising Initialised Quiescing Quiesced Terminated

Offset Hex	Type	Len	Name (Dim)	Description
(64)	UNSIGNED	4	NO_COLLECTIONS	# collections in CICS run
(68)	UNSIGNED	4	NO_SMF_WRITES	# SMF writes / interval
(6C)	FULLWORD	4	LENGTH_ DATA_WRITTEN	Len. data written / int.
(70)	CHARACTER	8	NEXT_COLL_EOD	EOD time used for next collection time calculation
(78)	FULLWORD	4	*	Reserved
(7C)	CHARACTER	8	CICS_START_TIME	CICS start time (STCK)

If USS records arrive during statistics collection they are chained for later processing.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	174	USS_BUFFER	
(0)	FULLWORD	4	UB_LENGTH	Length of whole buffer
(4)	FULLWORD	4	UB_DATA_LEN	Length of USS data only
(8)	CHARACTER	8	UB_CHAINING	
(8)	ADDRESS	4	UB_PREV	Previous and next in
(C)	ADDRESS	4	UB_NEXT	USS_CHAIN_PTR chain
(10)	CHARACTER	44	UB_SMF_HEADER	
(3C)	CHARACTER	114	UB_SMF_PS	
(AE)	CHARACTER		UB_DATA	Statistics data

STUCB Statistics utility program anchor block

STUP anchor block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	2375	ANCHOR	Anchor Block
(0)	CHARACTER	16	ANC_PREFIX	Anchor prefix area
(0)	HALFWORD	2	ANC_LENGTH	Anchor length
(2)	CHARACTER	1	ANC_ARROW	Arrow eyecatcher
(3)	CHARACTER	3	ANC_DFH	DFH
(6)	CHARACTER	2	ANC_DOMID	Domain id
(8)	CHARACTER	8	ANC_BLOCK_NAME	Control block name
(10)	CHARACTER	9	APPLID_SELECT (120)	
(10)	CHARACTER	8	APPLID	Applid selected
(18)	UNSIGNED	1	APPLID_FLAGS	Flags used in selection *
	1...		APPLID_ STATS_FOUND	Set when stats found on SMF for the applid
	.1..		*	unused
	..1.		*	unused
	...1		*	unused
	.... 1...		*	unused
	.... .1..		*	unused
	.... ..1.		*	unused
	1		*	unused
(448)	UNSIGNED	4	NUM_APPLID_ SELECT	Number selected
(44C)	CHARACTER	8	APPLID_IGNORE (120)	
(44C)	CHARACTER	8	APPLID	Applid ignored
(80C)	UNSIGNED	4	NUM_APPLID_ IGNORE	Number ignored
(810)	ADDRESS	4	APPLID_STATS_PTR	-> to applid statistics

```
  2 APPLID_STATS(40), Stats for report
    3 STATS_APPLID CHAR(8), Applid associated
    3 STATS_INTERVALS BIN(31), Interval count fo
    3 STATS_USSES BIN(31), Number of USS rec
    3 STATS_DATES (2) CHAR(8), First and last SM
                              dates - respectiv
    3 STATS_TIMES (2) CHAR(6), First and last SM
                              times - respectiv
```

Offset Hex	Type	Len	Name (Dim)	Description
(814)	CHARACTER	39	WRITE_PARMS	
(814)	UNSIGNED	2	PAGESIZE	Pagesize for report
(816)	UNSIGNED	2	LINES_WRITTEN	Lines written on current pg
(818)	UNSIGNED	2	PAGE_NUMBER	Page number so far
(81A)	CHARACTER	8	COLL_APPLID	Applid being reported
(822)	CHARACTER	8	COLL_JOBNAME	Jobname
(82A)	CHARACTER	6	COLL_TIME	Collection time
(830)	CHARACTER	8	COLL_DATE	Collection date
(838)	CHARACTER	3	STATS_COLL_TYPE	Coll type - INT/EOD/REQ/RRT/USS
(83B)	BITSTRING	1	REPORT_ REQD_FLAGS	

Offset Hex	Type	Len	Name (Dim)	Description
	1...		ALL	All reports produced
	.1..		EOD	End-of-day reports produced *
	..1.		INT	Interval reports produced *
	...1		REQ	Requested reports produced *
	.... 1...		USS	USS reports produced
	.... .1..		SUM	Summary report produced *
	.... ..1.		RRT	RRT reports produced
	1		*	Reserved
(83C)	CHARACTER	2	*	Reserved
(83E)	CHARACTER	8	CURRENT_APPLID	Applid being formatted
(848)	FULLWORD	4	CURRENT_INTERVAL	Interval no being formatted
(84C)	CHARACTER	8	CURRENT_DATE	yyyymmdd being formatted *
(854)	CHARACTER	6	CURRENT_TIME	hhmmss being formatted
(85A)	CHARACTER	8	CURRENT_ REQ_TOKEN	token for REQ report
(862)	CHARACTER	3	CURRENT_ REPORT_TYPE	type of report formatted *
(865)	CHARACTER	1	*	Reserved
(866)	HALFWORD	2	CURRENT_ PASS_NUMBER	pass currently executing *
(868)	HALFWORD	2	CURRENT_ NUM_APPLID	Number of applids found
(86A)	CHARACTER	40	CURRENT_ RESOURCE_ID	Resource ID being for'tted *
(892)	CHARACTER	2	CURRENT_ RECORD_TYPE	Record type being for'tted *
(894)	CHARACTER	2	*	Reserved
(898)	ADDRESS	4	CURRENT_ ENTRY_POINT	-> current format routine *
(89C)	CHARACTER	8	CURRENT_ CICS_START_TIME	Current start time STCK *
(8A4)	ADDRESS	4	SUMMARY_REC_PTR	-> to summary record
(8A8)	FULLWORD	4	SUMMARY_ REC_LENGTH	size of summary record
(8AC)	ADDRESS	4	TOTAL_REC_PTR	-> to total record
(8B0)	FULLWORD	4	TOTAL_REC_LENGTH	size of total record
(8B4)	ADDRESS	4	SUM_TOT_REC_PTR	-> to summary total record *
(8B8)	FULLWORD	4	SUM_TOT_ REC_LENGTH	size of summary total rec *
(8BC)	ADDRESS	4	SORT_RECORD_PTR	-> to sort record
(8C0)	FULLWORD	4	SORT_RECORD_LEN	size of sort record
(8C4)	ADDRESS	4	SMF_REC_PTR	-> to the SMF record
(8C8)	ADDRESS	4	SMF_REC_INDEX	-> to stats rec within SMF *
(8CC)	ADDRESS	4	STUP_KERNEL_PTR	-> to kernel stack entry *
(8D0)	ADDRESS	4	EXT_ENTRY_ TAB_PTR	-> to ext entry pt table *
(8D4)	CHARACTER	8	REPORT_DATE	mmddyyyy - report date
(8D4)	CHARACTER	2	REPORT_MM	
(8D6)	CHARACTER	2	REPORT_DD	
(8D8)	CHARACTER	4	REPORT_YYYY	
(8DC)	CHARACTER	6	REPORT_TIME	hhmmss - report time
(8DC)	CHARACTER	2	REPORT_HOUR	
(8DE)	CHARACTER	2	REPORT_MIN	
(8E0)	CHARACTER	2	REPORT_SEC	
(8E2)	CHARACTER	2	*	Filler
(8E4)	UNSIGNED	1	STATUS_FLAGS	
	1...		SMF_EMPTY	Flags an empty SMF log
	.1..		FIRST_INPUT_ RECORD	Flags the first input rec *
	..1.		FIRST_OUTPUT_ RECORD	Flags the first output rec *
	...1		COLLECT_STATS	Collect report stats
	.... 1...		WRITING_SUMMARY	Writing summary report
	.... .1..		WRITING_ REPORT_SUMM	Writing report summary
	.... ..1.		TIME_PERIOD_ SELECTED	A time period is selected
	1		TIME_PERIOD	Times are elapsed|daily
(8E5)	CHARACTER	3	STATS_FILE_OPEN	Stats file open flag
(8E8)	CHARACTER	16	REPORT_COUNTS	
(8E8)	FULLWORD	4	SMF_RECORD_ COUNT	Number of SMF records read *
(8EC)	FULLWORD	4	CICS_RECORD_ COUNT	Number of CICS records read *
(8F0)	FULLWORD	4	STATS_RECORD_ COUNT	Number of stats recs read *
(8F4)	FULLWORD	4	STATS_SELECTED_ COUNT	No. of stats recs selected *
(8F8)	CHARACTER	1	CURRENT_VERSION	Current stats dsect ver no. *
(8F9)	CHARACTER	8	CURRENT_ INTERVAL_TIME	Interval duration
(901)	CHARACTER	1	OTHER_SWITCHES	
	1...		UPPERCASE_REQ	Translate flag
	.111 1111		*	Filler
(902)	CHARACTER	2	*	Filler
(904)	ADDRESS	4	DFHMEBME_ADDR	Addr of DFHMEBM entry pt *
(908)	ADDRESS	4	MSG_TABLE_ADDR	Addr of message table
	Time/Date stamps for selected time period.			
(90C)	CHARACTER	14	SELECTED_PERIOD (2)	Row 1 = Start time/date Row 2 = Stop time/date

Offset Hex	Type	Len	Name (Dim)	Description
(90C)	CHARACTER	6	SELECTED_ TIME_PERIOD	Col 1 = Time - HHMMSS
(912)	CHARACTER	8	SELECTED_DATE_PERIOD	Col 2 = Date - MMDDYYYY *
(928)	CHARACTER	6	COLL_LAST_RESET	Last reset time

FORMATTER_FLAGS: Each formatter is invoked with one record at a time. If the current record read indicates that a reset of 'not reset' fields has occurred (i.e. CICS shutdown/cancel or USS records) then the RESET_OCCURRED bit os set ON.

Offset Hex	Type	Len	Name (Dim)	Description
(92E)	UNSIGNED	1	FORMATTER_FLAGS	flags for use by formatters *
	1...		RESET_OCCURRED	Reset occurred on prev. recd
	.1..		DFHSTWRK_ERROR_FLAG	Error with DFHSTWRK
	..11 1111		*	Reserved

SELECT_TYPE_FLAGS: Records can either be selected or ignored by the user by specifying SELECT/IGNORE TYPE input cards. If no selection is made, the default is to print all.

Offset Hex	Type	Len	Name (Dim)	Description
(92F)	BITSTRING	4	SELECT_ TYPE_FLAGS	Print selection flags
(92F)	CHARACTER	1	SELECT_ TYPE_FLAG1	
	1...		SELECT_ IGNORE_F	Select/ignore found
	.1..		SELECT_ AUTOINST	Select Autoinstall
	..1.		SELECT_CONNECT	Select Connection
	...1		SELECT_ DISPATCH	Select Dispatcher
	.... 1...		*	Reserved
	.... .1..		SELECT_FILE	Select File
	.... ..1.		SELECT_ LOGSTREAM	Select Logstream
	1		SELECT_JOURNAL	Select Journal
(930)	CHARACTER	1	SELECT_ TYPE_FLAG2	
	1...		SELECT_LSRPOOL	Select Lsrpool
	.1..		SELECT_MONITOR	Select Monitor
	..1.		SELECT_PROGRAM	Select Program
	...1		SELECT_STATS	Select Stats
	.... 1...		SELECT_STORAGE	Select Storage
	.... .1..		SELECT_SYSDUMP	Select Sysdump
	.... ..1.		SELECT_ TABLEMGR	Select Table Manager
	1		SELECT_ TCPIPSERVICE	Select TCPIP Services
(931)	CHARACTER	1	SELECT_ TYPE_FLAG3	
	1...		SELECT_TCLASS	Select Tclass
	.1..		SELECT_TDQUEUE	Select Tdqueue
	..1.		SELECT_ TERMINAL	Select Terminal
	...1		SELECT_ TRANDUMP	Select Trandump
	.... 1...		SELECT_ TRANSACT	Select Transaction
	.... .1..		SELECT_TSQUEUE	Select Tsqueue
	.... ..1.		SELECT_VTAM	Select Vtam
	1		SELECT_FEPI	Select FEPI
(932)	CHARACTER	1	SELECT_ TYPE_FLAG4	
	1...		SELECT_DBCTL	Select Dbcontrol
	.1..		SELECT_ PROGAUTO	Select Autoinstall program
	..1.		SELECT_DCE	Select DCE program
	...1		SELECT_USER	Select User domain
	.... 1...		*	Reserved
	.... .1..		SELECT_ENQUEUE	Select Enqueue
	.... ..1.		SELECT_ RECOVERY	Select Recovery
	1		SELECT_DB2	SELECT DB2
(933)	CHARACTER	20	PATCH_SPACE	Patch space

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	15000	STUP_APPLID_STATS	
(0)	CHARACTER	60	APPLID_STATS (250)	Statistics for report summary
(0)	CHARACTER	8	STATS_APPLID	Applid associated with statistics
(8)	FULLWORD	4	STATS_INTERVALS	Interval count for applid
(C)	FULLWORD	4	STATS_EODES	Number of EOD records
(10)	FULLWORD	4	STATS_INTES	Number of INT records
(14)	FULLWORD	4	STATS_REQES	Number of REQ records
(18)	FULLWORD	4	STATS_RRTES	Number of RRT records
(1C)	FULLWORD	4	STATS_USSES	Number of USS records
(20)	CHARACTER	8	STATS_DATES (2)	First and last SMF record dates - respectively
(30)	CHARACTER	6	STATS_TIMES (2)	First and last SMF record times - respectively

Constants

Len	Type	Value	Name	Description
2	DECIMAL	60	DEFAULT_PAGESIZE	
0	BIT	1	TRUE	
0	BIT	0	FALSE	
1	DECIMAL	1	STANDARD_PASS	
1	DECIMAL	2	SUMMARY_PASS	
4	DECIMAL	32769	BUFFER_LENGTH	
0	BIT	0	ELAPSED	
0	BIT	1	DAILY	

TIA Timer domain anchor block

CONTROL BLOCK NAME = DFHTIA
DESCRIPTIVE NAME = CICS Timer Domain (TI) Control Blocks
FUNCTION =
This file contains the data structure
declarations used by the Timer Domain.
The data structures are:
DFHTIA - TI Anchor block
TIMER_REQUEST_ELEMENT - TI Request Element
Notes:
Dependencies = S/370
Restrictions = none
Register Conventions = domain standard (no special usage)
Patch Label = N/A
Module Type = N/A
Attributes = N/A
TI domain Anchor Block storage definition

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	64	DFHTIA	Anchor block
(0)	CHARACTER	16	TIA_PREFIX	standard header
(0)	HALFWORD	2	TIA_LENGTH	length of anchor block
(2)	CHARACTER	1	TIA_ARROW	eyecatcher
(3)	CHARACTER	3	TIA_DFH	eyecatcher
(6)	CHARACTER	2	TIA_DOMID	domain id
(8)	CHARACTER	8	TIA_BLOCK_NAME	control block name
(10)	ADDRESS	4	TIA_LOCK_TOKEN	token required by Lock Manager
(14)	FULLWORD	4	TIA_SUSPEND_ TOKEN	token required by Dispatcher
(18)	FULLWORD	4	TIA_NUDGE_STATUS	DS nudge task state
(1C)	ADDRESS	4	TIA_DISPATCHER_ TOKEN	token to access dispatcher@P2A
(20)	CHARACTER	8	TIA_NEXT_ EXPIRY_TIME	next TRE expiry time
(20)	UNSIGNED	4	TIA_NEXT_ EXPIRY_HIGH	High-order word, stck secs@P2A
(24)	UNSIGNED	4	TIA_NEXT_ EXPIRY_LOW	Low-order word, stck usecs@P2A
(28)	CHARACTER	8	TIQC_SUBPOOL_ TOKEN	token required by SM on getmain
(30)	ADDRESS	4	TIA_FIRST_ TRE_PTR	-> head of the TRE chain
(34)	FULLWORD	4	TIA_REQUEST_ COUNTER	number of request notifies
(38)	BITSTRING	1	TIA_FLAGS	will need these
	1...		TIA_TIMER_ AVAILABLE	status bit for TI services
	.1..		*	unused
	..1.		*	unused
	...1		*	unused
	.... 1...		*	unused
	.... .1..		*	unused
	.... ..1.		*	unused
	1		*	unused
(39)	CHARACTER	3	*	reserved
(3C)	ADDRESS	4	KERR_PTR	-> Kernel recovery area

Timer Request Element Definition

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	84	TIMER_REQUEST_ ELEMENT	

Offset Hex	Type	Len	Name (Dim)	Description
				TRE
(0)	CHARACTER	24	TRE_PREFIX	standard header
(0)	HALFWORD	2	TRE_LENGTH	length of anchor block
(2)	CHARACTER	1	TRE_ARROW	eyecatcher
(3)	CHARACTER	3	TRE_DFH	eyecatcher
(6)	CHARACTER	2	TRE_DOMID	domain id
(8)	CHARACTER	8	TRE_BLOCK_NAME	control block name
(10)	ADDRESS	4	TRE_NEXT	-> next TRE in chain
(14)	ADDRESS	4	TRE_PREV	-> prev TRE in chain
(18)	FULLWORD	4	TRE_DOMAIN_ID	Number assigned by the Kernel
(1C)	CHARACTER	8	TRE_DOMAIN_TOKEN	Token from requesting domain
(1C)	UNSIGNED	4	*	
(20)	UNSIGNED	4	*	
(24)	CHARACTER	8	TRE_EXPIRY_TIME	Doubleword binary (STCK) time
(24)	UNSIGNED	4	TRE_EXPIRY_ TIME_HIGH	High-order word, stck secs
(28)	UNSIGNED	4	TRE_EXPIRY_ TIME_LOW	Low-order word, stck microsecs
(2C)	CHARACTER	8	TRE_INTERVAL	Doubleword binary interval
(2C)	UNSIGNED	4	TRE_INTERVAL_ SECS	Top 32 bytes contains seconds
(30)	UNSIGNED	4	TRE_INTERVAL_ MSECS	Bottom 32 bytes - microseconds
(34)	CHARACTER	6	TRE_ALARM_TIME	in HHMMSS format, local time
(3A)	CHARACTER	6	TRE_ORIGIN_TIME	HHMMSS, origin time of interval
(40)	CHARACTER	8	TRE_ORIGIN_DATE	MMDDYYYY, origin date of interval
(48)	UNSIGNED	1	TRE_NOTIFY_TYPE	type of notify requested
	1...		TRE_ALARM_CALL	Notify at certain time of day
	.1..		TRE_INTERVAL_ NOTIFY	notify after an interval
	..1.		TRE_ATTACHED_ TASK	notify via an attached task
	...1		TRE_TIMER_TASK	notify as part of timer thread
	.... 1...		TRE_PERIODIC	notify repeatedly
	.... .1..		TRE_WITH_ORIGIN	notify specified with an origin
	.... ..1.		TRE_WITH_ TIMEOUT	notify specified with a timeout
	1		TRE_WITH_ ATTMODE	notify specified with attach mode
(49)	UNSIGNED	1	TRE_FLAGS	various flags
	1...		TRE_EXPIRED	Expired, and notify in progress
	.1..		TRE_CANCELLED	Is it cancelled?
	..1.		TRE_ORIGIN_ INTERVAL_EXPIRED	expiry of 1st interval
	...1		TRE_RESET_ TIME_PROCESSED	local times adjusted?
	.... 1...		*	unused
	.... .1..		*	unused
	.... ..1.		*	unused
	1		*	unused
(4A)	CHARACTER	1	TRE_ATTACH_ PRIORITY	priority of task to be attached
(4B)	UNSIGNED	1	TRE_ATTACH_MODE	TCB mode of attached task
	1...		TRE_QR	Quasi-reentrant
	.1..		TRE_RO	Resource-owning
	..1.		TRE_CO	Concurrent
	...1		TRE_FO	File owning
(4C)	UNSIGNED	4	TRE_ATTACH_ TIMEOUT	attached notify timeout value
(50)	FULLWORD	4	TRE_NUMBER	request number for ttoken

Constants

Len	Type	Value	Name	Description
2	HEX	0001	TPID_TIDM_ENTRY	DFHTIDM entry
2	HEX	0002	TPID_TIDM_EXIT	DFHTIDM exit
2	HEX	0050	TPID_TIDM_INVDC	bad domain call
2	HEX	0051	TPID_TIDM_INVFMT	bad format number
2	HEX	0060	TPID_TIDM_RECOV	recovery routine
2	HEX	0100	TPID_TISR_ENTRY	DFHTISR entry
2	HEX	0101	TPID_TISR_EXIT	DFHTISR exit
2	HEX	0150	TPID_TISR_INVDC	bad domain call
2	HEX	0151	TPID_TISR_INVFMT	bad format number
2	HEX	0152	TPID_TISR_XINTVL	bad interval
2	HEX	0153	TPID_TISR_XTOKEN	bad token
2	HEX	0154	TPID_TISR_TOOLATE	TOD too late
2	HEX	0160	TPID_TISR_RECOV	recovery routine
2	HEX	0161	TPID_TISR_BADSTCK	MVS STCK problem
2	HEX	0162	TPID_TISR_NOATTACH	can't attach task
	Messages			
4	DECIMAL	1	MEID_RECOV	general abend
4	DECIMAL	4	MEID_LOOP	loop
4	DECIMAL	5	MEID_BADSTCK	stck inoperative
	Dumpcodes			

Len	Type	Value	Name	Description
8	CHARACTER	TI0001	DUID_TI_RECOV	general abend
8	CHARACTER	TI0004	DUID_TI_LOOP	loop
8	CHARACTER	TI0005	DUID_TI_BADSTCK	stck inoperative

Constants

1	CHARACTER	>	ARROW	eyectacher arrow
0	BIT	1	ON	TRUE flag value
0	BIT	0	OFF	FALSE flag value
0	BIT	1	YES	TRUE flag value
0	BIT	0	NO	FALSE flag value
7	CHARACTER	DFHTIDM	TIDM_NAME	module name
7	CHARACTER	DFHTISR	TISR_NAME	module name
4	HEX	FFFF0000	DELTA_ROUND	to zero low 2 bytes

TSA Temporary storage anchor block

-

TS domain anchor block, catalog record, constants and trace points.

TSA - TS Anchor block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	116	TSA	
(0)	CHARACTER	16	TSA_PREFIX	
(0)	HALFWORD	2	TSA_LENGTH	control block length
(2)	CHARACTER	1	TSA_ARROW	'>'
(3)	CHARACTER	3	TSA_DFH	'DFH'
(6)	CHARACTER	2	TSA_DOMID	'TS'
(8)	CHARACTER	8	TSA_BLOCK_NAME	'ANCHOR'
(10)	CHARACTER	8	TSA_TSGENRAL_ SPTOKEN	tsgenral subpool token
(18)	ADDRESS	4	TSA_TSNAME_ CLASSP	-> tsname class anchor
(1C)	ADDRESS	4	TSA_TSQUEUE_ CLASSP	-> tsqueue class anchor
(20)	ADDRESS	4	TSA_TSMAIN_ CLASSP	-> tsmain class anchor
(24)	ADDRESS	4	TSA_TSWAITQ_ CLASSP	-> tswaitq class anchor
(28)	ADDRESS	4	TSA_TSOLOCK_ CLASSP	-> tsolock class anchor
(2C)	ADDRESS	4	TSA_TSRLOCK_ CLASSP	-> tsrlock class anchor
(30)	ADDRESS	4	TSA_TSLOCK	TS domain global lock
(34)	ADDRESS	4	TSA_TSAUX_CLASSP	-> tsaux class anchor
(38)	UNSIGNED	1	TSA_TS_STATE	TS domain state
(39)	UNSIGNED	1	TSA_START	start type (see below)
(3A)	BITSTRING	1	TSA_FLAGS	flags
	1...		TSA_MAIN_ONLY	main-only support
	.1..		TSA_XTSQRIN_ ACTIVE	xtsqrin exit active
	..1.		TSA_XTSQROUT_ ACTIVE	xtsqrout exit active
	...1		TSA_XTSPTIN_ ACTIVE	xtsptin exit active
	.... 1...		TSA_XTSPTOUT_ ACTIVE	xtsptout exit active
	.... .1..		TSA_XRSINDI_ ACTIVE	xrsindi exit active
	.... ..1.		TSA_RDO_ENABLED	RDO for TST available
	1		*	reserved
(3B)	CHARACTER	1	*	reserved
(3C)	ADDRESS	4	TSA_TSTP	-> TST (or 0)
(40)	CHARACTER	8	TSA_LAST_ COLD_START_TIME	last cold start time
(48)	FULLWORD	4	TSA_BUFFERS	number of buffers
(4C)	FULLWORD	4	TSA_STRINGS	number of strings
(50)	CHARACTER	8	TSA_STATS_ RESET_TIME	time stats last reset
(58)	ADDRESS	4	TSA_SHARED_ ANCHORP	-> shared TS anchor block
(5C)	ADDRESS	4	TSA_SYSID_ TABLE_TOKEN	-> shared sysid table
(60)	CHARACTER	8	TSA_AGING_TIME	age queues created before this time
(68)	ADDRESS	4	TSA_TSMODEL_ CLASSP	-> tsmodel class anchor
(6C)	ADDRESS	4	*	reserved
(70)	ADDRESS	4	*	reserved
(74)	CHARACTER		*	

XMAT attach parms for CTSD delete recoverable queue transaction

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	24	CTSD_ATTACH_PARMS	
(0)	CHARACTER	16	CTSD_QUEUE_NAME	
(10)	CHARACTER	8	CTSD_LASTREF_ TIME	

Catalog record.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	CAT	
(0)	BITSTRING	1	CAT_FLAGS	
	1...		CAT_START_COLD	='1'b, cold start requested
	.111 1111		*	
(1)	CHARACTER	3	*	reserved
(4)	FULLWORD	4	CAT_BUFFERS	number of buffers requested
(8)	FULLWORD	4	CAT_STRINGS	number of strings requested
(C)	CHARACTER		*	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	1	TSA_START_COLD	
4	DECIMAL	2	TSA_START_WARM	
4	DECIMAL	3	TSA_START_ EMERGENCY	
4	DECIMAL	4	TSA_START_AUTO	
8	CHARACTER	TSDOMAIN	CAT_TYPE	
8	CHARACTER	TSSTATE	CAT_NAME	
Constants.				
2	CHARACTER	TS	COMPID	
8	CHARACTER	TSLOCK	TSLOCK_NAME	
1	CHARACTER	>	ARROW	
3	CHARACTER	DFH	DFH	
4	DECIMAL	3	DEFAULT_BUFFERS	
4	DECIMAL	3	DEFAULT_STRINGS	
SM domain states.				
4	DECIMAL	1	INITIALISING	
4	DECIMAL	2	INITIALISED	
4	DECIMAL	3	QUIESCING	
4	DECIMAL	4	QUIESCED	
4	DECIMAL	5	TERMINATED	
Standard message numbers and system dumpcode values.				
4	DECIMAL	1	MNO_ABEND	
8	CHARACTER	TS0001	DCD_ABEND	
4	DECIMAL	2	MNO_SEVERE_ERROR	
8	CHARACTER	TS0002	DCD_SEVERE_ERROR	
Non-standard message numbers.				
4	DECIMAL	100	MNO_INITIALISATION_STARTED	
4	DECIMAL	101	MNO_INITIALISATION_ENDED	
4	DECIMAL	102	MNO_FORMATTING_DATASET	
4	DECIMAL	103	MNO_INVALID_RDO_SWITCH	
Trace point id's.				
2	HEX	0101	TID_TSDM_ENTRY	
2	HEX	0102	TID_TSDM_EXIT	
2	HEX	0103	TID_TSDM_RECOVERY	
2	HEX	0104	TID_TSDM_INVALID_FORMAT	
2	HEX	0105	TID_TSDM_INVALID_FUNCTION	
2	HEX	0201	TID_TSQR_ENTRY	
2	HEX	0202	TID_TSQR_EXIT	
2	HEX	0203	TID_TSQR_RECOVERY	
2	HEX	0204	TID_TSQR_INVALID_FORMAT	

Len	Type	Value	Name	Description
2	HEX	0205	TID_TSQR_INVALID_FUNCTION	
2	HEX	0206	TID_TSQR_UNLOCK_ERROR_RECOVERY	
2	HEX	0301	TID_TSPT_ENTRY	
2	HEX	0302	TID_TSPT_EXIT	
2	HEX	0303	TID_TSPT_RECOVERY	
2	HEX	0304	TID_TSPT_INVALID_FORMAT	
2	HEX	0305	TID_TSPT_INVALID_FUNCTION	
2	HEX	0306	TID_TSPT_UNLOCK_ERROR_RECOVERY	
2	HEX	0401	TID_TSRM_ENTRY	
2	HEX	0402	TID_TSRM_EXIT	
2	HEX	0403	TID_TSRM_RECOVERY	
2	HEX	0404	TID_TSRM_INVALID_FORMAT	
2	HEX	0405	TID_TSRM_RMRO_INVALID_FUNCTION	
2	HEX	0406	TID_TSRM_RMDE_INVALID_FUNCTION	
2	HEX	0407	TID_TSRM_RMKP_INVALID_FUNCTION	
2	HEX	0408	TID_TSRM_UNLOCK_ERROR_RECOVERY	
2	HEX	0409	TID_TSRM_TSIC_INVALID_FUNCTION	
2	HEX	040A	TID_TSRM_QUEUE_RECOVERY_ ERR1	
2	HEX	040B	TID_TSRM_QUEUE_RECOVERY_ ERR2	
2	HEX	040C	TID_TSRM_SECTION_RECOVERY_ERR1	
2	HEX	040D	TID_TSRM_SECTION_RECOVERY_ERR2	
2	HEX	040E	TID_TSRM_SECTION_RECOVERY_ERR3	
2	HEX	040F	TID_TSRM_INVALID_LOG_RECORD	
2	HEX	0410	TID_TSRM_INV_INDOUBT_OPERATION	
2	HEX	0501	TID_TSST_ENTRY	
2	HEX	0502	TID_TSST_EXIT	
2	HEX	0503	TID_TSST_RECOVERY	
2	HEX	0504	TID_TSST_INVALID_FORMAT	
2	HEX	0505	TID_TSST_INVALID_FUNCTION	
2	HEX	0506	TID_TSST_UNLOCK_ERROR_RECOVERY	
2	HEX	0507	TID_TSST_STATS_BUFFER_TOO_SMALL	
2	HEX	0601	TID_TSSR_ENTRY	
2	HEX	0602	TID_TSSR_EXIT	
2	HEX	0603	TID_TSSR_RECOVERY	
2	HEX	0604	TID_TSSR_INVALID_FORMAT	
2	HEX	0605	TID_TSSR_INVALID_FUNCTION	
2	HEX	0606	TID_TSSR_UNLOCK_ERROR_RECOVERY	
2	HEX	0607	TID_TSSR_INVALID_EXIT_POINT	
2	HEX	0701	TID_TSBR_ENTRY	
2	HEX	0702	TID_TSBR_EXIT	
2	HEX	0703	TID_TSBR_RECOVERY	
2	HEX	0704	TID_TSBR_INVALID_FORMAT	
2	HEX	0705	TID_TSBR_INVALID_FUNCTION	
2	HEX	0706	TID_TSBR_UNLOCK_ERROR_RECOVERY	
2	HEX	0801	TID_TSWQ_ENTRY	
2	HEX	0802	TID_TSWQ_EXIT	
2	HEX	0803	TID_TSWQ_RECOVERY	
2	HEX	0804	TID_TSWQ_INVALID_FORMAT	
2	HEX	0805	TID_TSWQ_INVALID_FUNCTION	
2	HEX	0806	TID_TSWQ_UNLOCK_ERROR_RECOVERY	

Len	**Type**	**Value**	**Name**	**Description**
2	HEX	0807	TID_TSWQ_ DSSR_INQUIRE_SUSPEND	
2	HEX	0808	TID_TSWQ_ BEFORE_SUSPEND	
2	HEX	0809	TID_TSWQ_ AFTER_SUSPEND	
2	HEX	0901	TID_TSAM_ENTRY	
2	HEX	0902	TID_TSAM_EXIT	
2	HEX	0903	TID_TSAM_RECOVERY	
2	HEX	0904	TID_TSAM_ INVALID_FORMAT	
2	HEX	0905	TID_TSAM_ INVALID_FUNCTION	
2	HEX	0906	TID_TSAM_ 1310_ABEND_1	
2	HEX	0907	TID_TSAM_ 1310_ABEND_2	
2	HEX	0908	TID_TSAM_ 1310_ABEND_3	
2	HEX	0909	TID_TSAM_ 1310_ABEND_4	
2	HEX	090A	TID_TSAM_ 1310_ABEND_5	
2	HEX	090B	TID_TSAM_ 1310_ABEND_6	
2	HEX	090C	TID_TSAM_ 1310_ABEND_7	
2	HEX	090D	TID_TSAM_ 1310_ABEND_8	
2	HEX	090E	TID_TSAM_ 1310_ABEND_9	
2	HEX	090F	TID_TSAM_ 1310_ABEND_10	
2	HEX	0910	TID_TSAM_ 1310_ABEND_11	
2	HEX	0A01	TID_TSSH_ENTRY	
2	HEX	0A02	TID_TSSH_EXIT	
2	HEX	0A03	TID_TSSH_RECOVERY	
2	HEX	0A04	TID_TSSH_ INVALID_FORMAT	
2	HEX	0A05	TID_TSSH_ INVALID_FUNCTION	
2	HEX	0A06	TID_TSSH_ UNLOCK_ERROR_RECOVERY	
2	HEX	0A07	TID_TSSH_ BEFORE_CONNECT	
2	HEX	0A08	TID_TSSH_ AFTER_CONNECT	
2	HEX	0A09	TID_TSSH_ BEFORE_QUERY_SERVER	
2	HEX	0A0A	TID_TSSH_ AFTER_QUERY_SERVER	
2	HEX	0A0B	TID_TSSH_ BEFORE_SERVER_ REQUEST	
2	HEX	0A0C	TID_TSSH_ AFTER_SERVER_REQUEST	
2	HEX	0A0D	TID_TSSH_ BEFORE_CLOSE	
2	HEX	0A0E	TID_TSSH_ AFTER_CLOSE	
2	HEX	0B01	TID_TSAD_ENTRY	
2	HEX	0B02	TID_TSAD_EXIT	
2	HEX	0B03	TID_TSAD_RECOVERY	
2	HEX	0B04	TID_TSAD_ INVALID_FORMAT	
2	HEX	0B05	TID_TSAD_ INVALID_FUNCTION	
2	HEX	0B06	TID_TSAD_ UNLOCK_ERROR_RECOVERY	
2	HEX	0C01	TID_TSMB_ENTRY	
2	HEX	0C02	TID_TSMB_EXIT	
2	HEX	0C03	TID_TSMB_RECOVERY	
2	HEX	0C04	TID_TSMB_ INVALID_FORMAT	
2	HEX	0C05	TID_TSMB_ INVALID_FUNCTION	
2	HEX	0C06	TID_TSMB_ UNLOCK_ERROR_RECOVERY	
2	HEX	F701	TID_TSP_ENTRY	
2	HEX	F702	TID_TSP_EXIT	
2	HEX	F703	TID_TSP_INVALID_ REQUEST	
2	HEX	F704	TID_EITS_ENTRY	
2	HEX	F705	TID_EITS_EXIT	
2	HEX	F706	TID_EITS_RECOVERY	
2	HEX	F707	TID_EITS_ INVALID_FORMAT	
2	HEX	F708	TID_EITS_ INVALID_FUNCTION	
2	HEX	F709	TID_EITS_ INVALID_TS_FUNCTION	
2	HEX	F711	TID_TSDQ_ENTRY	
2	HEX	F712	TID_TSDQ_EXIT	
2	HEX	F713	TID_TSDQ_ERROR	

TSAUX Temporary storage auxiliary class

-

TSAUX class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	4	TSAUX	
INSTANCE DATA				
Declared Data				
(0)	CHAR Private	4	*	
ACA - aux control area.				
SHARED DATA				
Declared Data				
(0)	CHAR Protected	364	ACA	
(0)	CHAR Protected	16	ACA_PREFIX	
(0)	SIGNED Protected	2	ACA_LENGTH	control block length
(2)	CHAR Protected	1	ACA_ARROW	'>'
(3)	CHAR Protected	3	ACA_DFH	'DFH'
(6)	CHAR Protected	2	ACA_DOMID	'TS'
(8)	CHAR Protected	8	ACA_BLOCK_NAME	'ACA'
(10)	CHAR Protected	8	ACA_TSX_SPTOKEN	tstsx subpool token
(18)	CHAR Protected	8	ACA_TSS_SPTOKEN	tstss subpool token
(20)	CHAR Protected	8	ACA_TSBUFFER_ SPTOKEN	tsbuffer subpool token
(28)	OBJECT IsA(TSWAITQ) Protected	8	ACA_AUX_ SPACE_QUEUE	aux space wait queue
TSW - TS wait queue head.				
(28)	CHAR Protected	8	TSW_HEAD	
(28)	ADDRESS Protected	4	TSW_FIRST	-> first wait queue element
(2C)	ADDRESS Protected	4	TSW_LAST	-> last wait queue element
(30)	OBJECT IsA(TSWAITQ) Protected	8	ACA_EXTEND_ QUEUE	extend wait queue
(30)	CHAR Protected	8	TSW_HEAD	
(30)	ADDRESS Protected	4	TSW_FIRST	-> first wait queue element
(34)	ADDRESS Protected	4	TSW_LAST	-> last wait queue element
(38)	OBJECT IsA(TSWAITQ) Protected	8	ACA_BUFFER_ QUEUE	buffer wait queue
(38)	CHAR Protected	8	TSW_HEAD	
(38)	ADDRESS Protected	4	TSW_FIRST	-> first wait queue element
(3C)	ADDRESS Protected	4	TSW_LAST	-> last wait queue element
(40)	OBJECT IsA(TSWAITQ) Protected	8	ACA_WRITE_ BUFFER_QUEUE	write buffer queue
(40)	CHAR Protected	8	TSW_HEAD	
(40)	ADDRESS Protected	4	TSW_FIRST	-> first wait queue element
(44)	ADDRESS Protected	4	TSW_LAST	-> last wait queue element
(48)	OBJECT IsA(TSWAITQ) Protected	8	ACA_STRING_ QUEUE	string wait queue
(48)	CHAR Protected	8	TSW_HEAD	
(48)	ADDRESS Protected	4	TSW_FIRST	-> first wait queue element
(4C)	ADDRESS Protected	4	TSW_LAST	-> last wait queue element
(50)	ADDRESS Protected	4	ACA_ACBP	-> ts dataset acb
(54)	ADDRESS Protected	4	ACA_OPENLISTP	-> dataset open list
(58)	SIGNED Protected	4	ACA_OPENLIST_ LENGTH	length of open list

Offset Hex	Type	Len	Name (Dim)	Description
(5C)	ADDRESS Protected	4	ACA_OPENSKELP	-> open list skeleton
(60)	ADDRESS Protected	4	ACA_MODEL_RPLP	-> model rpl
(64)	SIGNED Protected	4	ACA_MAX_CIS_FORMATTED	maximum ci's formatted
(68)	ADDRESS Protected	4	ACA_FORMAT_ BUFFERP	-> buffer while formatting
(6C)	SIGNED Protected	4	ACA_FORMAT_RBA	-> rba while formatting
(70)	BITSTRING Protected	4	ACA_FORMAT_ECB	ecb while formatting
(74)	SIGNED Protected	4	ACA_NBCA	number of bcas
(78)	SIGNED Protected	4	ACA_NVCA	number of vcas
(7C)	SIGNED Protected	4	ACA_BLKN	number of bcas locked
(80)	SIGNED Protected	4	ACA_VLKN	number of vcas locked
(84)	ADDRESS Protected	4	ACA_BCAHD	-> first bca
(88)	ADDRESS Protected	4	ACA_BCAHA	-> first allocated bca
(8C)	ADDRESS Protected	4	ACA_BCAHF	-> first free bca
(90)	ADDRESS Protected	4	ACA_VCAHD	-> first vca
(94)	SIGNED Protected	4	ACA_RREFN	"read" reference number
(98)	SIGNED Protected	2	ACA_MAXWB	maximum write buffers
(9A)	SIGNED Protected	2	ACA_CURWB	current write buffers
(9C)	ADDRESS Protected	4	*	reserved
(A0)	ADDRESS Protected	4	*	reserved
(A4)	ADDRESS Protected	4	*	reserved
(A8)	ADDRESS Protected	4	*	reserved
Statistics fields.				
(AC)	CHAR Protected	60	ACA_STATS	
(AC)	SIGNED Protected	4	ACA_TRDN	total ci read count
(B0)	SIGNED Protected	4	ACA_TWTN	total ci write count
(B4)	SIGNED Protected	4	ACA_TWTNR	writes forced by recovery
(B8)	SIGNED Protected	4	ACA_TWTNF	formatting writes
(BC)	SIGNED Protected	4	ACA_NCIA	number of ci's allocated
(C0)	SIGNED Protected	4	ACA_NCIAH	hwm ci's allocated
(C4)	SIGNED Protected	4	ACA_NVCAH	hwm vcas alloc (strings)
(C8)	SIGNED Protected	4	ACA_VWTN	number of waits on vca
(CC)	SIGNED Protected	4	ACA_VUWT	no. users waiting on string
(D0)	SIGNED Protected	4	ACA_VUWTH	hwm users waiting on string
(D4)	SIGNED Protected	4	ACA_NAG	number of aux gets
(D8)	SIGNED Protected	4	ACA_BWTN	number of buffer waits
(DC)	SIGNED Protected	4	ACA_BUWT	users waiting for buffer
(E0)	SIGNED Protected	4	ACA_BUWTH	hwm users waiting for bufr
(E4)	SIGNED Protected	4	ACA_LAR	longest aux record len
Statistics fields which were in TS common area. (Old TSMxxx names are shown).				
(E8)	CHAR Protected	28	ACA_STATS2	
(E8)	SIGNED Protected	4	ACA_NP	(tsmsta1f) total records PUT (main/aux)
(EC)	SIGNED Protected	4	ACA_NPQ	(tsmsta2f) total records PUTQ (main/aux)

Offset Hex	Type	Len	Name (Dim)	Description
(F0)	SIGNED Protected	4	ACA_NAP	(tsmsta7f) total records PUT/Q aux
(F4)	SIGNED Protected	4	ACA_NSUSP	(tsmsta8f) number of suspensions
(F8)	SIGNED Protected	4	ACA_NCOMP	(tsmsta9f) number of compressions
(FC)	SIGNED Protected	4	ACA_NIOER	(tsmstaaf) number of I/O errors
(100)	SIGNED Protected	4	ACA_PGCSA	(tsmstabf) number of puts > ci size
(104)	SIGNED Protected	4	ACA_CSA	control interval size
(108)	SIGNED Protected	4	ACA_NCI	number of ci's
(10C)	SIGNED Protected	4	ACA_NAVB	num available bytes in ci
(110)	SIGNED Protected	4	ACA_BCID	displ. to buffer cntl info
(114)	SIGNED Protected	4	ACA_SPCI	segments per ci
(114)	CHAR Protected	3	*	padding for..
(117)	CHAR Protected	1	ACA_SPCI1	byte version of above
(118)	SIGNED Protected	4	ACA_BPSEG	bytes per seg
(11C)	SIGNED Protected	4	ACA_BPSG2	bytes per seg (as power 2)
Byte map pointers etc.				
(120)	ADDRESS Protected	4	ACA_BMP	-> byte map storage
(124)	ADDRESS Protected	4	ACA_MAPP	-> ts ci byte map
(128)	ADDRESS Protected	4	ACA_MAPEP	-> end of byte map
(12C)	ADDRESS Protected	4	ACA_SSP	start scan pointer
Controls for extending byte map.				
(130)	BITSTRING Protected	1	*	flags
	1... .... Protected		ACA_FULL	='1'b, dataset is full
	.1.. .... Protected		ACA_EXTENDING	='1'b, extension in progress
	..11 1111 Protected		*	reserved
(131)	CHAR Protected	3	*	reserved
(134)	SIGNED Protected	4	ACA_BMLEN	byte map length
(138)	SIGNED Protected	4	ACA_FTIME	time in binary seconds last "full" msg produced
(13C)	SIGNED Protected	4	ACA_FNCI	no. of ci's in dataset when last "full" msg produced
Fields set in the event of a 1310 abend.				
(140)	ADDRESS Protected	4	ACA_BCAP	-> bca for buffer being compressed
(144)	CHAR Protected	4	*	
(144)	SIGNED Protected	2	ACA_ASEGS	allocated segs (from ci)
(146)	SIGNED Protected	2	ACA_BSEGS	allocated segs (from map)
Fields used by 1310 trap.				
(148)	BITSTRING Protected	1	ACA_TRAP_FLAGS	trap flags
	1... .... Protected		ACA_COMPARE_ FAILED	='1', byte map copy failed
	.111 1111 Protected		*	reserved
(149)	CHAR Protected	3	*	reserved
(14C)	ADDRESS Protected	4	ACA_COPIED_BMP	-> copied byte map
(150)	ADDRESS Protected	4	*	reserved
(154)	ADDRESS Protected	4	*	reserved
(158)	ADDRESS Protected	4	*	reserved
(15C)	ADDRESS Protected	4	*	reserved
(160)	ADDRESS Protected	4	*	reserved
(164)	ADDRESS Protected	4	*	reserved
(168)	ADDRESS Protected	4	*	reserved
(16C)	CHAR Protected		*	
BCA - buffer control area.				

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	56	BCA	
(0)	CHAR Protected	8	BCA_NAPO	(for offset to bca_nap/nfp)
(0)	SIGNED Protected	2	BCA_LEN	length of this entry
(2)	BITSTRING Protected	1	BCA_FLAGS	flags:
	1... .... Protected		BCA_TBW	='1'b, to-be-written
	.1.. .... Protected		BCA_LOCK	='1'b, buffer is locked
	..1. .... Protected		BCA_RECOV	='1'b, recoverable data written to buffer
	...1 .... Protected		BCA_WBUF	='1'b, write buffer
	.... 1111 Protected		*	reserved
(3)	UNSIGNED Protected	1	*	reserved
(4)	ADDRESS Protected	4	BCA_CHNP	-> next buffer control area
(8)	CHAR Protected	48	*	
(8)	ADDRESS Protected	4	BCA_NAP	-> next allocated bca
(8)	ADDRESS Protected	4	BCA_NFP	-> next free bca
(C)	ADDRESS Protected	4	BCA_BUFP	-> buffer
(10)	ADDRESS Protected	4	BCA_NASP	-> next available segment
(14)	SIGNED Protected	4	BCA_CIN	ci number (0 when buffer is empty)
(18)	SIGNED Protected	4	BCA_WCIN	ci number for write opns
(1C)	SIGNED Protected	4	BCA_RREFN	read reference number
(20)	ADDRESS Protected	4	BCA_LR13	-> lock owners R13
(24)	SIGNED Protected	4	BCA_RDN	number of reads
(28)	SIGNED Protected	4	BCA_WTN	number of writes
(2C)	ADDRESS Protected	4	BCA_NLP	-> next locked buffer
(30)	UNSIGNED Protected	1	BCA_CIB	segs in cin (from map)
(31)	UNSIGNED Protected	1	BCA_WCIB	segs in wcin(from map)
(32)	CHAR Protected	2	*	reserved
(34)	SIGNED Protected	4	*	reserved
(38)	CHAR Protected		*	

Bytes in byte map for ci and write ci in a bca.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	1	CIB	
(0)	CHAR Protected	1	WCIB	

VCA - VSWA control area.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	20	VCA	
(0)	SIGNED Protected	2	VCA_LEN	length of this block
(2)	BITSTRING Protected	1	VCA_FLAGS	flags:
	1... .... Protected		VCA_LOCK	='1'b, VCA is locked
	.1.. .... Protected		VCA_IOP	='1'b, I/O in progress
	..11 1111 Protected		*	reserved
(3)	CHAR Protected	1	*	reserved
(4)	ADDRESS Protected	4	VCA_CHNP	-> next VSWA control area
(8)	BITSTRING Protected	4	VCA_ECB	ECB for VSAM to post
(C)	SIGNED Protected	4	VCA_RBA	RBA field
(10)	ADDRESS Protected	4	VCA_VSWAP	-> VSWA
(14)	CHAR Protected		*	

CTL - TS dataset control record.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	8	CTL	
(0)	CHAR Protected	8	CTL_NAME	control record name field
(8)	CHAR Protected		*	

BCI - buffer control information.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	11	BCI	
(0)	UNSIGNED Protected	1	*	reserved
(1)	UNSIGNED Protected	1	BCI_NASN	next available segment no.
(2)	SIGNED Protected	2	BCI_CINR	records in ci
(4)	CHAR Protected	7	BCI_RDF	RDF information (for VSAM)
(4)	CHAR Protected	1	*	reserved

Offset Hex	Type	Len	Name (Dim)	Description
(5)	UNSIGNED Protected	2	BCI_RDFSG	segment
(7)	UNSIGNED Protected	2	BCI_RDFRE	free
(9)	CHAR Protected	2	*	reserved
(B)	CHAR Protected		*	

BMH - byte map header.

(0)	CHAR Protected	16	BMH	
(0)	CHAR Protected	16	BMH_PREFIX	
(0)	SIGNED Protected	4	BMH_LENGTH	control block length
(4)	CHAR Protected	1	BMH_ARROW	'>'
(5)	CHAR Protected	3	BMH_DFH	'DFH'
(8)	CHAR Protected	2	BMH_DOMID	'TS'
(A)	CHAR Protected	6	BMH_BLOCK_NAME	'BMAP'
(10)	CHAR Protected		BMH_MAP_START	start of byte map

BMP - byte map.

(0)	UNSIGNED Protected	1	BMP (*)	
(0)	CHAR Protected	4	LLBB	
(0)	UNSIGNED Protected	2	LL	
(2)	UNSIGNED Protected	2	BB	
(0)	CHAR Protected	8	TSIOA	
(0)	CHAR Protected	8	TSIOA_EYECATCHER	

SLR - section log record.

(0)	CHAR Protected	44	SLR	
(0)	SIGNED Protected	2	SLR_LENGTH	record length
(2)	SIGNED Protected	2	SLR_PREV_OFFSET	offset to previous
(4)	CHAR Protected	4	SLR_RECORD_TYPE	'>TSS'
(8)	CHAR Protected	16	SLR_QUEUE_NAME	queue name
(18)	CHAR Protected	8	SLR_TIME_STAMP	time stamp
(20)	UNSIGNED Protected	2	SLR_ITEM_NUMBER	item number
(22)	UNSIGNED Protected	2	SLR_SECTION_ NUMBER	section number
(24)	UNSIGNED Protected	2	SLR_NUMBER_ OF_SECTIONS	number of sections
(26)	UNSIGNED Protected	2	SLR_TOTAL_ LENGTH	total item length
(28)	UNSIGNED Protected	2	SLR_CI_NUMBER	control interval number
(2A)	UNSIGNED Protected	2	SLR_SECTION_ LENGTH	length of this section
(2C)	CHAR Protected		*	

TSX - aux item descriptor.

(0)	CHAR Protected	16	TSX	
(0)	CHAR Protected	8	TSX_TIME_STAMP	item time stamp
(8)	SIGNED Protected	4	TSX_TOTAL_ LENGTH	total item length
(C)	ADDRESS Protected	4	TSX_TSSP	-> first TSS

TSS - aux section descriptor.

(0)	CHAR Protected	8	TSS	
(0)	ADDRESS Protected	4	TSS_NEXT	-> next TSS (or 0)
(4)	UNSIGNED Protected	2	TSS_CI_NUMBER	CI number
(6)	UNSIGNED Protected	2	TSS_SECTION_ LENGTH	length of section data

XRH - aux record header.

(0)	CHAR Protected	36	XRH	
(0)	SIGNED Protected	4	XRH_LENGTH	length of record (including header)
(4)	UNSIGNED Protected	2	XRH_ITEM_NUMBER	item number
(6)	UNSIGNED Protected	2	XRH_SECTION_ NUMBER	section number
(8)	CHAR Protected	8	XRH_TIME_STAMP	item time stamp
(10)	CHAR Protected	16	XRH_QUEUE_NAME	queue name
(20)	BITSTRING Protected	1	XRH_FLAGS	flags

Offset Hex	Type	Len	Name (Dim)	Description
	`1... ....` Protected		XRH_FMH	record has FMH
	`.1.. ....` Protected		XRH_RECOVERABLE	queue is recoverable
	`..1. ....` Protected		XRH_REQUIRED	record is required (used during buffer compression)
	`...1 1111` Protected		*	reserved
(21)	CHAR Protected	1	*	reserved
(22)	UNSIGNED Protected	2	XRH_SECTION_ LENGTH	data length of this section
(24)	CHAR Protected		XRH_DATA	start of section data
(0)	FIXED Public	4	TSX_RESPONSE	

Constants

Len	Type	Value	Name	Description
8	CHARACTER	ACA	ACA_BLOCK_ NAME_STRING	
8	CHARACTER	DFHTEMP	CTL_NAME_STRING	
6	CHARACTER	BMAP	BMH_BLOCK_ NAME_STRING	
8	CHARACTER	>TSIOA	TSIOA_EYECATCHER_ STRING	
4	DECIMAL	256	ZBMEXVAL	
Miscellaneous constants.				
4	DECIMAL	0	ZEMPTY	ci number for empty buffer
4	DECIMAL	1	ZMINREF	minimum ref no for a buffer
4	DECIMAL	0	TSX_OK	
4	DECIMAL	1	TSX_DISASTER	
4	DECIMAL	2	TSX_PURGED	
4	DECIMAL	3	TSX_NOSPACE	
4	DECIMAL	4	TSX_CHECK_FAILED	
4	DECIMAL	3	TSX_OPEN_FAILED	
4	DECIMAL	4	TSX_DATASET_EMPTY	
4	DECIMAL	5	TSX_CLOSE_FAILED	
4	DECIMAL	6	TSX_SHOWCB_FAILED	
4	DECIMAL	7	TSX_NO_CONTROL_ RECORD	

TSMN Temporary storage model class

-

TSMODEL class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	4	TSMODEL	
INSTANCE DATA				
Declared Data				
(0)	CHAR Private	4	*	
MDA - TS model class anchor block.				
SHARED DATA				
Declared Data				
(0)	CHAR Protected	44	MDA	
(0)	CHAR Protected	8	MDA_EYECATCHER	'>TSMDA '
(8)	CHAR Protected	8	MDA_MDB_SPTOKEN	mdb subpool token
(10)	CHAR Protected	8	MDA_MBR_SPTOKEN	mbr subpool token
(18)	CHAR Protected	8	MDA_MDBHEAD	
(18)	ADDRESS Protected	4	MDA_MDB_FIRST	-> first mdb
(1C)	ADDRESS Protected	4	MDA_MDB_LAST	-> last mdb
(20)	CHAR Protected	8	MDA_MBRHEAD	
(20)	ADDRESS Protected	4	MDA_MBR_FIRST	-> first mbr
(24)	ADDRESS Protected	4	MDA_MBR_LAST	-> last mbr
(28)	ADDRESS Protected	4	MDA_DEFAULT_ MDBP	-> default mdb
(2C)	CHAR Protected		*	
MDB - TS model block.				
(0)	CHAR Protected	120	MDB	
(0)	CHAR Protected	8	MDB_MDBHEAD	chain fields
(0)	ADDRESS Protected	4	MDB_NEXT	-> next mdb
(4)	ADDRESS Protected	4	MDB_PREV	-> previous mdb
(8)	STRUCTURE IsA(TSMODELNAME) Protected	8	MDB_NAME	model name field
(10)	CHAR Protected	16	MDB_QNAME	queue name field
(20)	STRUCTURE IsA(TSPREFIX) Protected	16	MDB_PREFIX	prefix (as input)
(30)	CHAR Protected	16	MDB_PREFIX_MASK	prefix mask (0s for wild)
(40)	STRUCTURE IsA(TSPREFIX) Protected	16	MDB_MASKED_ PREFIX	mask and-ed with prefix
(50)	SIGNED Protected	4	MDB_PREFIXLEN	significant length of prefix
(54)	BITSTRING Protected	1	MDB_FLAGS	flags
	1... .... Protected		MDB_MAIN	='1'b, main
	.1.. .... Protected		MDB_RECOVERABLE	='1'b, recoverable
	..1. .... Protected		MDB_SECURITY	='1'b, security
	...1 .... Protected		MDB_DEFAULT	='1'b, default mdb
	.... 1111 Protected		*	reserved
(55)	CHAR Protected	3	*	reserved
(58)	STRUCTURE IsA(POOLNAME) Protected	8	MDB_POOL_NAME	pool name
(60)	ADDRESS Protected	4	MDB_POOL_TOKEN	pool token
(64)	STRUCTURE IsA(TSSYSID) Protected	4	MDB_SYSID	sysid
(68)	STRUCTURE IsA(TSPREFIX) Protected	16	MDB_REMOTE_ PREFIX	remote prefix
(78)	CHAR Protected		*	
MBR - tsmodel browse block.				
(0)	CHAR Protected	52	MBR	
(0)	CHAR Protected	8	MBR_MBRHEAD	chain fields
(0)	ADDRESS Protected	4	MBR_NEXT	-> next mbr

Offset Hex	Type	Len	Name (Dim)	Description
(4)	ADDRESS Protected	4	MBR_PREV	-> previous mbr
(8)	CHAR Protected	4	MBR_TRANID	browsing tranid
(C)	CHAR Protected	4	MBR_TRANNUM	browsing tran number
(10)	CHAR Protected	8	MBR_TRANTOKEN	browsing tran token
(18)	STRUCTURE IsA(TSPREFIX) Protected	16	MBR_PREFIX	current cursor value
(28)	SIGNED Protected	4	*	Reserved (was change count).
(2C)	ADDRESS Protected	4	*	Reserved (was -> current mdb)
(30)	ADDRESS Protected	4	*	reserved
(0)	CHAR Public	8	TSMODELNAME	
(0)	CHAR Public	16	TSPREFIX	
(0)	CHAR Public	8	POOLNAME	
(0)	CHAR Public	4	TSSYSID	
(0)	FIXED Public	4	MDL_RESPONSE	

Constants

Len	Type	Value	Name	Description
1	CHARACTER	+	WILDCHAR	
1	CHARACTER		BLANK	
0	BIT	1	TRUE	
0	BIT	0	FALSE	
8	CHARACTER	>TSMDA	MDA_EYECATCHER_STRING	
8	CHARACTER	TSMODEL	TSMD_MODEL_TYPE	
8	CHARACTER	TSRDO4TS	TSMD_RDO_TYPE	
8	CHARACTER	STATUS	TSMD_RDO_NAME	
8	CHARACTER	ENABLED	TSMD_RDO_ENABLED	
8	CHARACTER	DISABLED	TSMD_RDO_DISABLED	
4	DECIMAL	0	MDL_OK	
4	DECIMAL	1	MDL_NOT_FOUND	
4	DECIMAL	2	MDL_DUPLICATE_NAME	
4	DECIMAL	3	MDL_DUPLICATE_ PREFIX	
4	DECIMAL	4	MDL_END_BROWSE	
4	DECIMAL	5	MDL_INVALID_PREFIX	
4	DECIMAL	6	MDL_PURGED	
4	DECIMAL	7	MDL_DISASTER	
4	DECIMAL	8	MDL_INVALID_NAME	
4	DECIMAL	9	MDL_INVALID_BROWSE_TOKEN	
4	DECIMAL	10	MDL_CATALOG_ERROR	

TSMN Temporary storage main class

-

TSMAIN class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	4	TSMAIN	
INSTANCE DATA				
Declared Data				
(0)	CHAR Private	4	*	

-

Note that set storage address/length a temporary for testing under CMS.

TSM - tsmain class anchor.

SHARED DATA				
Declared Data				
(0)	CHAR Protected	88	TSM_CLASS_ANCHOR	
(0)	SIGNED Protected	4	TSM_NMP	number main put/putq's
(4)	SIGNED Protected	4	TSM_NMG	number of main get/getq's
(8)	SIGNED Protected	4	TSM_CURV	current tsmain storage
(C)	SIGNED Protected	4	TSM_MAXV	peak tsmain storage
(10)	CHAR Protected	8	TSM_SPTOKEN (0 8)	fixed sp tokens
(58)	CHAR Protected		*	

TSM - main item header.

(0)	CHAR Protected	8	TSM	
(0)	CHAR Protected	8	TSM_PREFIX	
(0)	CHAR Protected	4	TSM_EYECATCHER	'>TSM'
(4)	BITSTRING Protected	2	TSM_FLAGS	flags
(4)	BITSTRING Protected	1	*	
	1... .... Protected		TSM_FMH	header in data
	.111 1111 Protected		*	reserved
(5)	CHAR Protected	1	*	reserved
(6)	UNSIGNED Protected	2	TSM_LENGTH	item data length
(8)	CHAR Protected		TSM_DATA	start of user data

LLBB - length header.

(0)	CHAR Protected	4	LLBB	
(0)	UNSIGNED Protected	2	LL	length
(2)	UNSIGNED Protected	2	BB	'0000'x

TSIOA - tsioa eyecatcher.

(0)	CHAR Protected	8	TSIOA	
(0)	CHAR Protected	8	TSIOA_EYECATCHER	

Fixed length subpool arrays.

(0)	SIGNED Protected	2	TSM_FIXED_ LENGTH_TAB (8)	
(10)	CHAR Protected	4	TSM_SUFFIX_TAB (8)	

--

(0)	FIXED Public	4	TSM_RESPONSE	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	8	FIXED_SUBPOOLS	
4	DECIMAL	64	FIXED_LENGTH_ MULTIPLE	
4	DECIMAL	64	VARIABLE_ SUBPOOL_BOUNDARY	
4	DECIMAL	512	FIXED_LENGTH_ MAXIMUM	
4	CHARACTER	TSMN	TSM_SPPREFIX	
4	CHARACTER	>TSM	TSM_EYECATCHER_ VALUE	
8	CHARACTER	>TSIOA	TSIOA_EYECATCHER_ STRING	
4	DECIMAL	0	TSM_OK	
4	DECIMAL	1	TSM_INVALID_ EYECATCHER	
4	DECIMAL	2	TSM_PURGED	
4	DECIMAL	3	TSM_DISASTER	

TSNM Temporary storage name class

-

TSNAME class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	4	TSNAME	
INSTANCE DATA				
Declared Data				
(0)	CHAR Private	4	*	

TSN - tsname class anchor block.

Offset Hex	Type	Len	Name (Dim)	Description
SHARED DATA				
Declared Data				
(0)	CHAR Protected	56	TSN_CLASS_ANCHOR	
(0)	ADDRESS Protected	4	TSN_ROOTP	-> root node
(4)	ADDRESS Protected	4	*	reserved
(8)	CHAR Protected	8	TSN_DTN_SPTOKEN	tsdtn subpool token
(10)	CHAR Protected	8	TSN_TSQ_SPTOKEN	tsqueue subpool token
(18)	CHAR Protected	8	TSN_BRB_SPTOKEN	tsbrb subpool token
(20)	CHAR Protected	12	*	statistics
(20)	SIGNED Protected	4	TSN_QNUM	number of queues
(24)	SIGNED Protected	4	TSN_QNUMH	peak number of queues
(28)	SIGNED Protected	4	TSN_NQCR	times queue created
(2C)	SIGNED Protected	4	TSN_CHANGE_ COUNT	directory change count
(30)	CHAR Protected	8	TSN_BRBHEAD	
(30)	ADDRESS Protected	4	TSN_BRB_FIRST	-> first browse block
(34)	ADDRESS Protected	4	TSN_BRB_LAST	-> last browse block
(38)	CHAR Protected		*	

DTN - digital tree node.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	88	DTN	
(0)	CHAR Protected	16	DTN_NAME	name field
(10)	ADDRESS Protected	4	DTN_UP	-> up node (or zero)
(14)	UNSIGNED Protected	1	DTN_OFFSET	offset to byte containing index digit
(15)	UNSIGNED Protected	1	DTN_SUBTRACT	value to subtract to isolate index digit
(16)	UNSIGNED Protected	1	DTN_SHIFT	shift value to islolate index digit
(17)	UNSIGNED Protected	1	DTN_DOWN_COUNT	count of non-zero down pointers
(18)	ADDRESS Protected	4	DTN_DOWN (0 15)	down pointer array
(58)	CHAR Protected		DTN_END	end of down pointer array

BRB - browse block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	56	BRB	
(0)	ADDRESS Protected	4	BRB_NEXT	-> next brb
(4)	ADDRESS Protected	4	BRB_PREV	-> previous brb
(8)	CHAR Protected	4	BRB_TRANID	browsing tranid
(C)	CHAR Protected	4	BRB_TRANNUM	browsing tran number
(10)	CHAR Protected	8	BRB_TRANTOKEN	browsing tran token
(18)	CHAR Protected	16	BRB_NAME	current name value
(28)	SIGNED Protected	4	BRB_CHANGE_ COUNT	change count at last get_next
(2C)	ADDRESS Protected	4	BRB_NODEP	-> current node
(30)	ADDRESS Protected	4	BRB_SLOTP	-> current slot within node
(34)	ADDRESS Protected	4	*	reserved
(0)	FIXED Public	4	TSN_RESPONSE	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	0	TSN_OK	
4	DECIMAL	1	TSN_NOT_FOUND	
4	DECIMAL	2	TSN_DUPLICATE	
4	DECIMAL	3	TSN_END_BROWSE	
4	DECIMAL	4	TSN_INVALID_PREFIX	
4	DECIMAL	5	TSN_PURGED	
4	DECIMAL	6	TSN_DISASTER	
4	DECIMAL	7	TSN_INVALID_NAME	

TSOL Temporary storage ownership lock class

-

TSOLOCK class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	4	TSOLOCK	

TSO - TS ownership lock.

INSTANCE DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	ADDRESS Protected	4	TSO_QOBP	-> queue ownership block

QAB - queue ownership anchor block.

SHARED DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	560	QAB	
(0)	CHAR Protected	8	QAB_PREFIX	
(0)	ADDRESS Protected	4	QAB_NEXT	-> next QAB
(4)	ADDRESS Protected	4	QAB_PREV	-> previous QAB
(8)	CHAR Protected	8	QAB_UOWID	UOW id
(10)	ADDRESS Protected	4	QAB_TASK_TOKEN	task token
(14)	CHAR Protected	4	QAB_TRANSACTION_ NUMBER	transaction number
(18)	CHAR Protected	8	QAB_QOBHEAD	
(18)	ADDRESS Protected	4	QAB_QOB_FIRST	-> first QOB
(1C)	ADDRESS Protected	4	QAB_QOB_LAST	-> last QOB
(20)	CHAR Protected	8	QAB_MDBHEAD	
(20)	ADDRESS Protected	4	QAB_MDB_FIRST	-> first MDB
(24)	ADDRESS Protected	4	QAB_MDB_LAST	-> last MDB

Offset Hex	Type	Len	Name (Dim)	Description
(28)	BITSTRING Protected	1	QAB_FLAGS	
	1... .... Protected		QAB_SHUNTED	UOW has been shunted
	.1.. .... Protected		QAB_UNSHUNTED	UOW has been unshunted
	..11 1111 Protected		*	
(29)	CHAR Protected	3	*	
(2C)	CHAR Protected	16	QAB_LOG_BUFFER_HEADER	
(3C)	CHAR Protected	500	QAB_LOG_BUFFER	
QOB - queue ownership block.				
(0)	CHAR Protected	44	QOB	
(0)	CHAR Protected	8	QOB_PREFIX	
(0)	ADDRESS Protected	4	QOB_NEXT	-> next QOB for this UOW
(4)	ADDRESS Protected	4	QOB_PREV	-> previous QOB for this UOW
(8)	CHAR Protected	16	QOB_QUEUE_NAME	queue name
(18)	OBJECT IsA(TSWAITQ) Protected	8	QOB_WAITQ	ownership wait queue
TSW - TS wait queue head.				
(18)	CHAR Protected	8	TSW_HEAD	
(18)	ADDRESS Protected	4	TSW_FIRST	-> first wait queue element
(1C)	ADDRESS Protected	4	TSW_LAST	-> last wait queue element
(20)	ADDRESS Protected	4	QOB_QABP	-> QAB
(24)	ADDRESS Protected	4	QOB_QTOKEN	queue token
(28)	ADDRESS Protected	4	QOB_NQTOKEN	enqueue token
(2C)	CHAR Protected		*	
TSO - tsolock class anchor block.				
(0)	CHAR Protected	2052	TSO_CLASS_ANCHOR	
(0)	CHAR Protected	8	TSO_QAB_SPTOKEN	qab subpool token
(8)	CHAR Protected	8	TSO_QOB_SPTOKEN	qob subpool token
(10)	ADDRESS Protected	4	TSO_NQTOKEN	enq pool token
(14)	ADDRESS Protected	4	*	reserved
(18)	CHAR Protected	8	TSO_QABHEAD	
(18)	ADDRESS Protected	4	TSO_QAB_FIRST	-> first qab
(1C)	ADDRESS Protected	4	TSO_QAB_LAST	-> last qab
(20)	ADDRESS Protected	4	*	reserved
(24)	CHAR Protected	16	TSO_KEYPT_BUFFER_HEADER	
(34)	CHAR Protected	2000	TSO_KEYPT_ BUFFER	
(804)	CHAR Protected		*	
LBH - log buffer header.				
(0)	CHAR Protected	16	LBH	
(0)	ADDRESS Protected	4	LBH_P	address of buffer
(4)	UNSIGNED Protected	4	LBH_N	length of data in buffer
(8)	SIGNED Protected	4	LBH_M	total length of buffer
(C)	SIGNED Protected	4	*	reserved
(0)	FIXED Public	4	TSO_RESPONSE	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	500	QAB_LOG_BUFFER_ LENGTH	
4	DECIMAL	2000	TSO_KEYPT_ BUFFER_LENGTH	
4	DECIMAL	0	TSO_OK	
4	DECIMAL	1	TSO_PURGED	
4	DECIMAL	2	TSO_DISASTER	
4	DECIMAL	3	TSO_RESTART	
4	DECIMAL	4	TSO_LOCKED	

TSQU Temporary storage queue class

-

TSQUEUE class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	128	TSQUEUE	
TSQ - TS queue control block.				
INSTANCE DATA				
Declared Data				
(0)	CHAR Protected	128	TSQ	
(0)	CHAR Protected	20	TSQ_PREFIX	
(0)	CHAR Protected	16	TSQ_NAME	queue name
(10)	ADDRESS Protected	4	TSQ_UP	-> "up" node
(14)	CHAR Protected	108	TSQ_REST	
(14)	ADDRESS Protected	4	TSQ_FIRST_TSIP	-> first TSI
(18)	ADDRESS Protected	4	TSQ_LAST_TSIP	-> last TSI
(1C)	SIGNED Protected	4	TSQ_TOTAL_ ITEMS	total items
(20)	SIGNED Protected	4	TSQ_READ_ CURSOR	read cursor
(24)	ADDRESS Protected	4	TSQ_READ_TSIP	-> read TSI
(28)	OBJECT IsA(TSRLOCK) Protected	16	TSQ_REQUEST_ LOCK	request lock
(28)	CHAR Protected	8	TSR_WAITQ	
TSW - TS wait queue head.				
(28)	CHAR Protected	8	TSW_HEAD	
(28)	ADDRESS Protected	4	TSW_FIRST	-> first wait queue element
(2C)	ADDRESS Protected	4	TSW_LAST	-> last wait queue element
TSR - TS queue request lock.				
(30)	ADDRESS Protected	4	TSR_OWNER	.
(38)	OBJECT IsA(TSOLOCK) Protected	4	TSQ_OWNERSHIP_ LOCK	ownership lock
TSO - TS ownership lock.				
(38)	ADDRESS Protected	4	TSO_QOBP	-> queue ownership block
(3C)	SIGNED Protected	4	TSQ_COMMITTED_ ITEMS	committed item count
(40)	CHAR Protected	8	TSQ_QUBHEAD	qub chain header
(40)	ADDRESS Protected	4	TSQ_QUB_FIRST	-> first QUB
(44)	ADDRESS Protected	4	TSQ_QUB_LAST	-> last QUB
(48)	CHAR Protected	8	TSQ_CREATION_ TIME	time created

Offset Hex	Type	Len	Name (Dim)	Description
(50)	CHAR Protected	8	TSQ_LAST_ REFERENCED_TIME	time last referenced
(58)	CHAR Protected	4	TSQ_TRANSID	creating transid
(5C)	ADDRESS Protected	4	TSQ_IC_DATA_P	-> ic data (or 0)
(60)	BITSTRING Protected	2	TSQ_FLAGS	(see below)
(62)	UNSIGNED Protected	1	TSQ_FIRST_ OPERATION	first operation ("put" queues only)
(63)	CHAR Protected	1	*	reserved
(64)	ADDRESS Protected	4	TSQ_OLD_ IC_DATA_P	-> old ice (or 0)
(68)	CHAR Protected	8	TSQ_OLD_ CREATION_TIME	creation time for backout
(70)	ADDRESS Protected	4	*	reserved
(74)	ADDRESS Protected	4	*	reserved
(78)	ADDRESS Protected	4	*	reserved
(7C)	ADDRESS Protected	4	*	reserved
(80)	CHAR Protected		*	

TSQ flags.

(60)	BITSTRING Public	2	TSQ_FLAG_BYTES	
(60)	BITSTRING Public	1	*	
	1... .... Protected		TSQ_MAIN	='1'b, queue is main
	.1.. .... Protected		TSQ_BMS	='1'b, queue owned by BMS
	..1. .... Protected		TSQ_IC	='1'b, queue owned by ICP
	...1 .... Protected		TSQ_PUT	='1'b, put-type queue
	.... 1... Protected		TSQ_RECOVERABLE	='1'b, queue recoverable
	.... .1.. Protected		TSQ_DELETED	='1'b, logically deleted
	.... ..1. Protected		TSQ_OWNED	='1'b, queue is owned
	....1 Protected		TSQ_SHUNTED	='1'b, queue is shunted
(61)	BITSTRING Public	1	*	
	1... .... Protected		TSQ_DISCARD	='1'b, will discard queue
	.1.. .... Protected		TSQ_NEW	='1'b, queue just created
	..1. .... Protected		TSQ_DELETE_ SEEN	='1'b, delete seen (log)
	...1 1111 Protected		*	reserved

TSI - TS item descriptor.

SHARED DATA
Declared Data

(0)	CHAR Protected	8	TSI	item descriptor
(0)	ADDRESS Protected	4	TSI_NEXT	-> next TSI
(4)	ADDRESS Protected	4	TSI_ITEMT	item token

QUB - queue update block.

(0)	CHAR Protected	20	QUB	queue update block
(0)	ADDRESS Protected	4	QUB_NEXT	-> next QUB
(4)	ADDRESS Protected	4	QUB_PREV	-> previous QUB
(8)	SIGNED Protected	4	QUB_ITEM_NUMBER	item number updated
(C)	ADDRESS Protected	4	QUB_OLD_ITEMT	before image token
(10)	ADDRESS Protected	4	QUB_TSIP	-> tsi for after image

TSQ - class anchor block.

(0)	CHAR Protected	36	TSQ_CLASS_ANCHOR	
(0)	CHAR Protected	8	TSQ_TSI_SPTOKEN	TSI subpool token
(8)	CHAR Protected	8	TSQ_QUB_SPTOKEN	QUB subpool token
(10)	CHAR Protected	8	TSQ_IC_SPTOKEN	TSICDATA subpool token
(18)	ADDRESS Protected	4	TSQ_TSIFREEHEAD	head of TSI free chain
(1C)	SIGNED Protected	4	TSQ_IC_DATA_N	length of ic_data items
(20)	SIGNED Protected	4	TSQ_QINH	items in longest queue
(24)	CHAR Protected		*	

QLR - queue type log record.

(0)	CHAR Protected	72	QLR	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	SIGNED Protected	2	QLR_LENGTH	block length
(2)	SIGNED Protected	2	QLR_PREV_OFFSET	offset to previous
(4)	CHAR Protected	4	QLR_RECORD_TYPE	'>TSQ'
(8)	CHAR Protected	16	QLR_QUEUE_NAME	queue name
(18)	CHAR Protected	8	QLR_CREATION_ TIME	creation time
(20)	CHAR Protected	8	QLR_LAST_ REFERENCED_TIME	last referenced
(28)	CHAR Protected	4	QLR_TRANSID	creating transid
(2C)	UNSIGNED Protected	2	QLR_TOTAL_ITEMS	total items in queue
(2E)	UNSIGNED Protected	2	QLR_COMMITTED_ ITEMS	total committed items
(30)	UNSIGNED Protected	2	QLR_READ_CURSOR	read cursor
(32)	BITSTRING Protected	2	QLR_FLAGS	flags
(32)	BITSTRING Public	1	*	
	1... .... Protected		TSQ_MAIN	
	.1.. .... Protected		TSQ_BMS	
	..1. .... Protected		TSQ_IC	
	...1 .... Protected		TSQ_PUT	
	.... 1... Protected		TSQ_RECOVERABLE	
	.... .1.. Protected		TSQ_DELETED	
	.... ..1. Protected		TSQ_OWNED	
	....1 Protected		TSQ_SHUNTED	
(33)	BITSTRING Public	1	*	
	1... .... Protected		TSQ_DISCARD	
	.1.. .... Protected		TSQ_NEW	
	..1. .... Protected		TSQ_DELETE_ SEEN	
	...1 1111 Protected		*	
(34)	CHAR Protected	1	QLR_FIRST_ OPERATION	first operation
(35)	CHAR Protected	1	*	reserved
(36)	UNSIGNED Protected	2	QLR_IC_DATA_N	length of any ic data
(38)	UNSIGNED Protected	2	QLR_OLD_ IC_DATA_N	length of any old ice
(3A)	UNSIGNED Protected	2	*	reserved
(3C)	SIGNED Protected	4	*	reserved
(40)	CHAR Protected	8	QLR_OLD_ CREATION_TIME	old create time
(48)	CHAR Protected		QLR_IC_DATA	start of any ic data
Response from tsqueue methods.				
(0)	FIXED Public	4	TSQ_RESPONSE	
Storage types.				
(0)	FIXED Public	1	STGTYPE	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	32767	MAXITEMS	maximum items in a queue
4	DECIMAL	32763	MAXITEMLENGTH	maximum item length
4	DECIMAL	0	TSQ_OPERATION_NULL	
4	DECIMAL	1	TSQ_OPERATION_PUT	
4	DECIMAL	2	TSQ_OPERATION_ GET_RELEASE	
4	DECIMAL	3	TSQ_OPERATION_ RELEASE	
4	DECIMAL	0	TSQ_OK	
4	DECIMAL	1	TSQ_DISASTER	
4	DECIMAL	2	TSQ_FULL	
4	DECIMAL	3	TSQ_ITEM_NOT_FOUND	
4	DECIMAL	4	TSQ_PURGED	
4	DECIMAL	5	TSQ_INVALID_LENGTH	
4	DECIMAL	6	TSQ_RESTART	
4	DECIMAL	7	TSQ_LOCKED	
4	DECIMAL	8	TSQ_QUEUE_DELETED	
4	DECIMAL	9	TSQ_NOSPACE	
4	DECIMAL	10	TSQ_CHECK_FAILED	
4	DECIMAL	11	TSQ_INVALID_TYPE	
4	DECIMAL	12	TSQ_DUPLICATE_NAME	
1	DECIMAL	1	STGTYPE_MAIN	
1	DECIMAL	2	STGTYPE_AUX_TST	

TSRL Temporary storage shared class

-

TSSHARED class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	4	TSSHARED	
INSTANCE DATA				
Declared Data				
(0)	CHAR Private	4	*	

SHA - tsshared class anchor block.

Offset Hex	Type	Len	Name (Dim)	Description
SHARED DATA				
Declared Data				
(0)	CHAR Protected	72	SHA	
(0)	CHAR Protected	16	SHA_PREFIX	
(0)	SIGNED Protected	2	SHA_LENGTH	control block length
(2)	CHAR Protected	1	SHA_ARROW	'>'
(3)	CHAR Protected	3	SHA_DFH	'DFH'
(6)	CHAR Protected	2	SHA_COMPID	'TS'
(8)	CHAR Protected	8	SHA_BLOCK_NAME	'SHA'

Note: The following level 2 structure is also used in DFHTSSHI.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHAR Protected	16	SHA_SYSID_TABLE	
(10)	CHAR Protected	8	SHA_STEHEAD	
(10)	ADDRESS Protected	4	SHA_STE_FIRST	-> first ste
(14)	ADDRESS Protected	4	SHA_STE_LAST	-> last ste
(18)	CHAR Protected	8	SHA_PCAHEAD	
(18)	ADDRESS Protected	4	SHA_PCA_FIRST	-> first pca
(1C)	ADDRESS Protected	4	SHA_PCA_LAST	-> last pca
(20)	CHAR Protected	8	SHA_SBBHEAD	
(20)	ADDRESS Protected	4	SHA_SBB_FIRST	-> first sbb
(24)	ADDRESS Protected	4	SHA_SBB_LAST	-> last sbb
(28)	CHAR Protected	8	SHA_PBBHEAD	
(28)	ADDRESS Protected	4	SHA_PBB_FIRST	-> first pbb
(2C)	ADDRESS Protected	4	SHA_PBB_LAST	-> last pbb
(30)	CHAR Protected	24	SHA_STATISTICS	
(30)	SIGNED Protected	4	SHA_POOLS_ DEFINED	number of pools defined
(34)	SIGNED Protected	4	SHA_POOLS_ CONNECTED	number of pools connected to
(38)	SIGNED Protected	4	SHA_READ_ REQUESTS	number of shared reads
(3C)	SIGNED Protected	4	SHA_WRITE_ REQUESTS	number of shared writes
(40)	SIGNED Protected	4	*	
(44)	SIGNED Protected	4	*	
(48)	CHAR Protected		*	

STE - sysid table entry.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	16	STE	
(0)	CHAR Protected	8	STE_PREFIX	
(0)	ADDRESS Protected	4	STE_NEXT	-> next ste
(4)	ADDRESS Protected	4	STE_PREV	-> previous ste
(8)	CHAR Protected	4	STE_SYSID	sysid
(C)	ADDRESS Protected	4	STE_PCAP	-> pca for this sysid

PCA - pool control area.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	32	PCA	
(0)	CHAR Protected	8	PCA_PREFIX	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	ADDRESS Protected	4	PCA_NEXT	-> next pca
(4)	ADDRESS Protected	4	PCA_PREV	-> previous pca
(8)	CHAR Protected	8	PCA_POOL_NAME	pool name
(10)	OBJECT IsA(TSWAITQ) Protected	8	PCA_WAIT_QUEUE	wait queue

TSW - TS wait queue head.

Offset Hex	Type	Len	Name (Dim)	Description
(10)	CHAR Protected	8	TSW_HEAD	
(10)	ADDRESS Protected	4	TSW_FIRST	-> first wait queue element
(14)	ADDRESS Protected	4	TSW_LAST	-> last wait queue element
(18)	ADDRESS Protected	4	PCA_CONNECT_ TOKEN	connect token
(1C)	BITSTRING Protected	1	PCA_FLAGS	
	1... .... Protected		PCA_CONNECT_ FAILED	='1'b, connect failed
	.111 1111 Protected		*	
(1D)	CHAR Protected	3	*	

SBB - shared browse block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	48	SBB	
(0)	CHAR Protected	8	SBB_PREFIX	
(0)	ADDRESS Protected	4	SBB_NEXT	-> next sbb
(4)	ADDRESS Protected	4	SBB_PREV	-> previous sbb
(8)	CHAR Protected	4	SBB_TRANID	browsing tranid
(C)	CHAR Protected	4	SBB_TRANNUM	browsing tran number
(10)	CHAR Protected	8	SBB_TRANTOKEN	browsing tran token
(18)	CHAR Protected	16	SBB_NAME	current browse name
(28)	ADDRESS Protected	4	SBB_PCAP	-> pool control area
(2C)	BITSTRING Protected	1	SBB_FLAGS	
	1... .... Protected		SBB_FIRST	='1'b, first get_next
	.111 1111 Protected		*	reserved
(2D)	CHAR Protected	3	*	reserved
(30)	CHAR Protected		*	

PBB - pool browse block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	32	PBB	
(0)	CHAR Protected	8	PBB_PREFIX	
(0)	ADDRESS Protected	4	PBB_NEXT	-> next pbb
(4)	ADDRESS Protected	4	PBB_PREV	-> previous pbb
(8)	CHAR Protected	4	PBB_TRANID	browsing tranid
(C)	CHAR Protected	4	PBB_TRANNUM	browsing tran number
(10)	CHAR Protected	8	PBB_TRANTOKEN	browsing tran token
(18)	CHAR Protected	8	PBB_POOL_NAME	current shared TS pool name
(20)	CHAR Protected		*	
(0)	FIXED Public	4	TSH_RESPONSE	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	32768	SETSTGL	
4	DECIMAL	0	TSH_OK	
4	DECIMAL	1	TSH_DISASTER	
4	DECIMAL	2	TSH_NOT_FOUND	
4	DECIMAL	3	TSH_PURGED	
4	DECIMAL	4	TSH_BROWSE_END	

TSRL Temporary storage resource lock class

-

TSRLOCK class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	16	TSRLOCK	
INSTANCE DATA				
Declared Data				
(0)	OBJECT IsA(TSWAITQ) Protected	8	TSR_WAITQ	
TSW - TS wait queue head.				
(0)	CHAR Protected	8	TSW_HEAD	
(0)	ADDRESS Protected	4	TSW_FIRST	-> first wait queue element
(4)	ADDRESS Protected	4	TSW_LAST	-> last wait queue element
TSR - TS queue request lock.				
(8)	ADDRESS Protected	4	TSR_OWNER	.
SHARED DATA				
Declared Data				
(0)	CHAR Protected	8	TSR_CLASS_ANCHOR	
(0)	CHAR Protected	8	*	reserved
(8)	CHAR Protected		*	
(0)	FIXED Public	4	TSR_RESPONSE	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	0	TSR_OK	
4	DECIMAL	1	TSR_DELETED	
4	DECIMAL	2	TSR_PURGED	
4	DECIMAL	3	TSR_DISASTER	
4	DECIMAL	4	TSR_RESTART	

TSWQ Temporary storage wait queue class

-

TSWAITQ class.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	8	TSWAITQ	

TSW - TS wait queue head.

INSTANCE DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	8	TSW_HEAD	
(0)	ADDRESS Protected	4	TSW_FIRST	-> first wait queue element
(4)	ADDRESS Protected	4	TSW_LAST	-> last wait queue element

TSW - TS wait queue element.

SHARED DATA
Declared Data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	31	TSW	
(0)	CHAR Protected	8	TSW_PREFIX	
(0)	ADDRESS Protected	4	TSW_NEXT	-> next wait queue element
(4)	ADDRESS Protected	4	TSW_PREV	-> prev wait queue element
(8)	ADDRESS Protected	4	TSW_SUSPEND_ TOKEN	suspend token
(C)	ADDRESS Protected	4	TSW_WAITER	waiter (task token)
(10)	CHAR Protected	8	TSW_SUSPEND_ START_TIME	suspend start time
(18)	CHAR Protected	4	TSW_TRANSACTION_ NUMBER	transaction number
(1C)	BITSTRING Protected	1	TSW_FLAGS	
	`1... .... Protected`		TSW_RESTART_ REQUIRED	='1'b, restart reqd
	`.111 1111 Protected`		*	
(1D)	FIXED Protected	1	TSW_RESOURCE_ TYPE	resource type
(1E)	UNSIGNED Protected	1	TSW_RESUME_ PRIORITY	resume priority
(1F)	CHAR Protected		*	
(0)	CHAR Public	8	TSW_CLASS_ANCHOR	
(0)	CHAR Public	8	TSW_TSW_SPTOKEN	tsw subpool token
(8)	CHAR Public		*	

Responses.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	FIXED Public	4	TSW_RESPONSE	

Resource types.
Note that these values must be kept in step with the resource_ type option on the append_ waiter function.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	FIXED Public	1	TSW_RESTYPE	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	0	TSW_OK	
4	DECIMAL	1	TSW_RESTART	
4	DECIMAL	2	TSW_PURGED	
4	DECIMAL	3	TSW_DISASTER	
1	DECIMAL	1	TSW_AUX_SPACE	
1	DECIMAL	2	TSW_BUFFER	
1	DECIMAL	3	TSW_WRITE_BUFFER	
1	DECIMAL	4	TSW_STRING	
1	DECIMAL	5	TSW_EXTEND	
1	DECIMAL	6	TSW_QUEUE	
1	DECIMAL	7	TSW_POOL	

UDB User domain user data block

-

DFHUSUDC US User Data Block

The UDB defines the operator data and user attributes associated with a user who has been added to the CICS system.

It is owned by the USAD Gate of the user domain.

It contains the non-security attributes of the user that have been obtained from the CICS and LANGUAGE segments in the External Security Manager's database. It also contains a pointer to the ACEE (Access Control Environment Element), but ONLY for the use of the EXEC CICS ADDRESS ACEE command. There are NO security capabilities contained in the UDB - only the External Security Manager has knowledge of these. If the User Data Block is enabled for timeout processing, then the user timeout queue entry (UTQE) token, which identifies the entry in the User Timeout Queue (UTQ), is stored in the user data block.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	120	USUD_USER_DATA	User Data Block
(0)	ADDRESS	4	USUD_USER_TOKEN	User Token
(4)	ADDRESS	4	USUD_UTQE_TOKEN	Token for timer queue
(8)	CHARACTER	8	USUD_SECURITY_ TOKEN	Security Token
(8)	ADDRESS	4	P	
(C)	FULLWORD	4	N	
(10)	FULLWORD	4	USUD_ADD_ USE_COUNT	ADD_USER use count
(14)	FULLWORD	4	USUD_TRAN_ USE_COUNT	Transaction use count
(18)	ADDRESS	4	USUD_ACEE_PTR	User's ACEE address
(1C)	HALFWORD	2	USUD_TIMEOUT_ INTERVAL	Timeout Interval (mins)
(1E)	BITSTRING	1	USUD_USER_ OPTIONS	User options
	1...		USUD_SCOPE_ CHECK	Apply SNSCOPE to user
	.1..		USUD_SCOPE_ OBTAINED	Scope ENQ obtained
	..1.		USUD_DELETE_ IMMEDIATE	Delete immedia
	...1 111.		*	Reserved
	1		USUD_XRF_ REFLECTABLE	Reflect signon to XRF
(1F)	CHARACTER	11	USUD_USERID	Userid of this user
(1F)	UNSIGNED	1	LEN	
(20)	CHARACTER	10	VAL	
(2A)	UNSIGNED	1	USUD_OPERATOR_ PRIORITY	Operator Priority
(2B)	CHARACTER	11	USUD_GROUPID	Groupid supplied
(2B)	UNSIGNED	1	LEN	
(2C)	CHARACTER	10	VAL	
(36)	CHARACTER	1	*	Reserved
(37)	CHARACTER	11	USUD_CURRENT_ GROUPID	

Offset Hex	Type	Len	Name (Dim)	Description
				Current Groupid
(37)	UNSIGNED	1	LEN	
(38)	CHARACTER	10	VAL	
(42)	CHARACTER	1	*	Reserved
(43)	CHARACTER	9	USUD_ENTRY_PORT	Port of Entry
(43)	UNSIGNED	1	TYPE	
(44)	CHARACTER	8	NAME	
(4C)	ADDRESS	4	USUD_USDDB_PTR	User's DDB address
(50)	CHARACTER	8	USUD_APPLID	Originating applid
(58)	CHARACTER	1	*	Reserved
(59)	CHARACTER	3	USUD_NATIONAL_ LANGUAGE	
				National Language
(5C)	BITSTRING	3	USUD_OPERATOR_ CLASSES	
				Operator Classes
(5C)	BITSTRING	1	USUD_OPCLASS_ BYTE (0 2)	Address individual bytes
(5F)	BITSTRING	1	*	Reserved
(60)	CHARACTER	20	USUD_USERNAME	Personal name of user
(74)	CHARACTER	1	*	Reserved
(75)	CHARACTER	3	USUD_OPERATOR_ IDENT	
				Operator Identifier
(78)	CHARACTER		*	End

-

User Directory

Define the directory key

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	USDK_DIRECTORY_ KEY	User Directory Key
(0)	CHARACTER	10	USDK_USERID	Userid
(A)	CHARACTER	3	USDK_SCOPE_ ACTIVE	Scope check required
(D)	CHARACTER	10	USDK_GROUPID	Groupid
(17)	CHARACTER	9	USDK_ENTRY_PORT	Entry Port
(17)	UNSIGNED	1	TYPE	
(18)	CHARACTER	8	NAME	
(20)	CHARACTER	8	USDK_APPLID	Applid
(28)	CHARACTER	16	USDK_UUID	Reserved for future use
(38)	CHARACTER		*	End

USANC User domain anchor block

-

DFHUSANC - User Domain Anchor Block

This anchor block contains the global storage for the user domain.

It defines the domain state information, variables and constants required by the US gates and other external programs such as DFHUSTRI, the user domain trace interpretation routine.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	168	USA	
(0)	CHARACTER	16	USA_PREFIX	Eyecatcher prefix
(0)	HALFWORD	2	USA_PREFIX_ LENGTH	Length of US anchor
(2)	CHARACTER	14	USA_PREFIX_TEXT	>DFHUSANCHOR
Domain state information				
(10)	UNSIGNED	1	USA_US_STATE	US domain state: initialized, quiesced or terminated
System initialization parameters and general flags				
(11)	UNSIGNED	1	USA_SIGNON_SCOPE	SNSCOPE (Signon scope)
(12)	BITSTRING	1	USA_FLAGS	General flags
	1...		USA_ENQ_ LIMIT_EXCEEDED_ MSG	ENQ limit message already issued.
	.111 1111		*	Spare flags
(13)	CHARACTER	4	*	Reserved
(17)	STRUCTURE IsA(USERID)	11	USA_DEFAULT_ USERID	DFLTUSER (Default userid)
(17)	UNSIGNED	1	LEN	
(18)	CHARACTER	10	VAL	
(22)	HALFWORD	2	*	Reserved
(24)	UNSIGNED	4	USA_DIRECTORY_ TIMEOUT_VALUE	USRDELAY (in TOD units)
(28)	CHARACTER	8	USA_GENERIC_ APPLID	Generic applid
Subpool Tokens				
(30)	STRUCTURE IsA(ETOKEN)	8	USA_GENERAL_ SPTOKEN	General subpool, including the anchor
(30)	ADDRESS	4	P	
(34)	FULLWORD	4	N	
(38)	STRUCTURE IsA(ETOKEN)	8	USA_XMTRAN_ SPTOKEN	Transaction data subpool
(38)	ADDRESS	4	P	
(3C)	FULLWORD	4	N	
(40)	STRUCTURE IsA(ETOKEN)	8	USA_USERDATA_ SPTOKEN	User data subpool
(40)	ADDRESS	4	P	
(44)	FULLWORD	4	N	
(48)	STRUCTURE IsA(ETOKEN)	8	USA_UTQE_SPTOKEN	Timeout queue subpool
(48)	ADDRESS	4	P	
(4C)	FULLWORD	4	N	
(50)	STRUCTURE IsA(ETOKEN)	8	USA_DCEDATA_ SPTOKEN	DCE data subpool
(50)	ADDRESS	4	P	
(54)	FULLWORD	4	N	
Pointers				
(58)	ADDRESS	4	USA_DEFAULT_ USUDB_PTR	Ptr to default user usudb
(5C)	ADDRESS	4	USA_USER_ TIMEOUT_QUEUE_PTR	Ptr to timeout queue
User Directory related data				
(60)	ADDRESS	4	USA_DIRKEY_ DIRECTORY_TOKEN	Userid directory
(64)	ADDRESS	4	USA_USERTOKEN_ DIRECTORY_TOKEN	Token directory

Offset Hex	Type	Len	Name (Dim)	Description
Tokens				
(68)	STRUCTURE IsA(ETOKEN)	8	USA_TIMER_TOKEN	Token from Timer Domain
(68)	ADDRESS	4	P	
(6C)	FULLWORD	4	N	
(70)	STRUCTURE IsA(ETOKEN)	8	USA_JOBSTEP_TRANS_TOKEN	Transaction token for jobstep user
(70)	ADDRESS	4	P	
(74)	FULLWORD	4	N	
(78)	ADDRESS	4	USA_DEFAULT_USER_TOKEN	DFLTUSER's token
(7C)	FULLWORD	4	USA_USER_ TOKEN_HWM	Token high water mark
(80)	ADDRESS	4	USA_LOCK_TOKEN1	US lock token 1
(84)	ADDRESS	4	USA_LOCK_TOKEN2	US lock token 2
Statistics				
(88)	UNSIGNED	4	USA_TIMEOUT_MEAN_REUSE_TIME	Average time to reuse
(8C)	UNSIGNED	4	USA_TIMEOUT_REUSE_COUNT	Number of reuses
(90)	UNSIGNED	4	USA_TIMEOUT_EXPIRY_COUNT	Number of expirys
(94)	UNSIGNED	4	USA_DIRECTORY_REUSE_COUNT	Number of reuses
(98)	UNSIGNED	4	USA_DIRECTORY_NOT_FOUND_COUNT	Number of not-founds
(9C)	CHARACTER	8	USA_LAST_ RESET_TIME	Statistics reset time
(A4)	CHARACTER	4	*	avoid silly compiler msgs
(A8)	CHARACTER		*	Reserved for alignment

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	US_STATE_ INITIALIZING	
1	DECIMAL	2	US_STATE_ INITIALIZED	
1	DECIMAL	3	US_STATE_QUIESCING	
1	DECIMAL	4	US_STATE_QUIESCED	
1	DECIMAL	5	US_STATE_ TERMINATED	
Signon Scope options				
1	DECIMAL	1	US_SCOPE_NONE	
1	DECIMAL	2	US_SCOPE_CICS	
1	DECIMAL	3	US_SCOPE_MVSIMAGE	
1	DECIMAL	4	US_SCOPE_SYSPLEX	
Component id (for use on ME domain calls)				
2	CHARACTER	US	COMPID	Used on ME domain calls
Standard message numbers and system dumpcode values				
1	DECIMAL	1	MNO_ABEND	
8	CHARACTER	US0001	DCD_ABEND	
1	DECIMAL	2	MNO_SEVERE_ERROR	
8	CHARACTER	US0002	DCD_SEVERE_ERROR	
1	DECIMAL	3	MNO_NO_STORAGE	
8	CHARACTER	US0003	DCD_NO_STORAGE	
1	DECIMAL	4	MNO_LOOP	
8	CHARACTER	US0004	DCD_LOOP	
1	DECIMAL	5	MNO_STCK_ERROR	
8	CHARACTER	US0005	DCD_STCK_ERROR	
1	DECIMAL	6	MNO_NO_MVS_STORAGE	
8	CHARACTER	US0006	DCD_NO_MVS_STORAGE	
1	DECIMAL	120	MNO_ENQ_LIMIT_EXCEEDED	
Trace Point Identifiers				
2	HEX	0101	TID_USDM_ENTRY	
2	HEX	0102	TID_USDM_EXIT	
2	HEX	0103	TID_USDM_RECOVERY	
2	HEX	0104	TID_USDM_INVALID_FORMAT	
2	HEX	0105	TID_USDM_INVALID_FUNCTION	
2	HEX	0106	TID_USDM_UNLOCK_ERROR	

Len	Type	Value	Name	Description
2	HEX	0107	TID_USDM_NO_STORAGE_FOR_ USA	
2	HEX	0108	TID_USDM_GET_PARMS_FAILED	
2	HEX	0201	TID_USIS_ENTRY	
2	HEX	0202	TID_USIS_EXIT	
2	HEX	0203	TID_USIS_RECOVERY	
2	HEX	0204	TID_USIS_ INVALID_FORMAT	
2	HEX	0205	TID_USIS_INVALID_FUNCTION	
2	HEX	0206	TID_USIS_NO_INQUIRE_PARAMETERS	
2	HEX	0207	TID_USIS_NO_SET_PARAMETERS	
2	HEX	0208	TID_USIS_ LOCK_ERROR	
2	HEX	0209	TID_USIS_ UNLOCK_ERROR	
2	HEX	020A	TID_USIS_UNLOCK_ERROR_RECOVERY	
2	HEX	0301	TID_USAD_ENTRY	
2	HEX	0302	TID_USAD_EXIT	
2	HEX	0303	TID_USAD_RECOVERY	
2	HEX	0304	TID_USAD_INVALID_FORMAT	
2	HEX	0305	TID_USAD_INVALID_FUNCTION	
2	HEX	0306	TID_USAD_ LOCK_ERROR	
2	HEX	0307	TID_USAD_UNLOCK_ERROR	
2	HEX	0308	TID_USAD_UNLOCK_ERROR_RECOVERY	
2	HEX	0309	TID_USAD_EXCEPTION_UNKNOWN	
2	HEX	030A	TID_USAD_EXTRACT_FAILED	
2	HEX	030B	TID_USAD_INVALID_PARAMETERS	
2	HEX	030C	TID_USAD_USER_NOT_IN_DIRECTORY	
2	HEX	030D	TID_USAD_USER_DIR_ADD_DUPLICATE	
2	HEX	030E	TID_USAD_USER_DIR_ADD_ERROR	
2	HEX	030F	TID_USAD_USER_DIR_DELETE_ ERROR	
2	HEX	0310	TID_USAD_INVALID_SECURITY_ TOKEN	
2	HEX	0311	TID_USAD_USE_COUNT_ERROR	
2	HEX	0312	TID_USAD_DFHUSER_DEQ_FAILED	
2	HEX	0313	TID_USAD_UDB_PTR_INVALID	
2	HEX	0314	TID_USAD_ADD_TIMEOUT_FAILED	
2	HEX	0315	TID_USAD_DEL_TIMEOUT_FAILED	
2	HEX	0316	TID_USAD_DEL_EXPIRED_FAILED	
2	HEX	0317	TID_USAD_INVALID_DCE_STATE	
2	HEX	0318	TID_USAD_DCE_EXCEPTION_UNKNOWN	
2	HEX	0401	TID_USXM_ENTRY	
2	HEX	0402	TID_USXM_EXIT	
2	HEX	0403	TID_USXM_RECOVERY	
2	HEX	0404	TID_USXM_INVALID_FORMAT	
2	HEX	0405	TID_USXM_INVALID_FUNCTION	
2	HEX	0406	TID_USXM_ LOCK_ERROR	
2	HEX	0407	TID_USXM_UNLOCK_ERROR	
2	HEX	0408	TID_USXM_UNLOCK_ERROR_RECOVERY	
2	HEX	0409	TID_USXM_GETMAIN_FAILURE	
2	HEX	040A	TID_USXM_DIRMAN_FAILURE	
2	HEX	040B	TID_USXM_TRAN_USE_COUNT_ MAX	
2	HEX	040C	TID_USXM_TRAN_USE_COUNT_ NEG	
2	HEX	040D	TID_USXM_TRAN_USE_COUNT_ LOW	
2	HEX	040E	TID_USXM_BAD_SECURITY_TOKEN	

Len	Type	Value	Name	Description
2	HEX	040F	TID_USXM_ TOKEN_TYPE_ERROR	
2	HEX	0410	TID_USXM_ INVALID_TRANSACTION_ TOKEN	
2	HEX	0411	TID_USXM_ ALREADY_ADDED_ SECURITY	
2	HEX	0412	TID_USXM_ NO_PRINCIPAL_UDB_ PTR	
2	HEX	0413	TID_USXM_ USAD_ERROR	
2	HEX	0501	TID_USFL_ENTRY	
2	HEX	0502	TID_USFL_EXIT	
2	HEX	0503	TID_USFL_RECOVERY	
2	HEX	0504	TID_USFL_ INVALID_FORMAT	
2	HEX	0505	TID_USFL_ INVALID_FUNCTION	
2	HEX	0506	TID_USFL_ LOCK_ERROR	
2	HEX	0507	TID_USFL_ UNLOCK_ERROR	
2	HEX	0508	TID_USFL_ UNLOCK_ERROR_RECOVERY	
2	HEX	0509	TID_USFL_ EXCEPTION_UNKNOWN	
2	HEX	050B	TID_USFL_ USER_NOT_IN_DIRECTORY	
2	HEX	050C	TID_USFL_ USER_DIR_ADD_DUPLICATE	
2	HEX	050D	TID_USFL_ UNFLATTEN_USER_ ERROR	
2	HEX	050E	TID_USFL_ USER_DIR_DELETE_ ERROR	
2	HEX	050F	TID_USFL_ INVALID_SECURITY_ TOKEN	
2	HEX	0510	TID_USFL_ USE_COUNT_ERROR	
2	HEX	0511	TID_USFL_ DFHUSER_DEQ_FAILED	
2	HEX	0512	TID_USFL_ UDB_PTR_INVALID	
2	HEX	0513	TID_USFL_ DEL_TIMEOUT_FAILED	
2	HEX	0601	TID_USST_ENTRY	
2	HEX	0602	TID_USST_EXIT	
2	HEX	0603	TID_USST_RECOVERY	
2	HEX	0604	TID_USST_ INVALID_FORMAT	
2	HEX	0605	TID_USST_ INVALID_FUNCTION	
2	HEX	0606	TID_USST_ LOCK_ERROR	
2	HEX	0607	TID_USST_ UNLOCK_ERROR	
2	HEX	0608	TID_USST_ UNLOCK_ERROR_RECOVERY	
2	HEX	0701	TID_USTI_ENTRY	
2	HEX	0702	TID_USTI_EXIT	
2	HEX	0703	TID_USTI_RECOVERY	
2	HEX	0704	TID_USTI_ INVALID_FORMAT	
2	HEX	0705	TID_USTI_ INVALID_FUNCTION	
2	HEX	0706	TID_USTI_ LOCK_ERROR	
2	HEX	0707	TID_USTI_ UNLOCK_ERROR	
2	HEX	0708	TID_USTI_ UNLOCK_ERROR_RECOVERY	
2	HEX	0709	TID_USTI_ EXCEPTION_UNKNOWN	
2	HEX	070A	TID_USTI_ UDB_PTR_INVALID	
2	HEX	070B	TID_USTI_ ADD_QUEUE_ENTRY_ ERROR	
2	HEX	070C	TID_USTI_ ALREADY_IN_QUEUE	
2	HEX	070D	TID_USTI_ DELETE_QUEUE_ENTRY_ ERROR	
2	HEX	070E	TID_USTI_ GET_QUEUE_ENTRY_ ERROR	
2	HEX	070F	TID_USTI_ QUEUE_ENTRY_IN_ USE	
2	HEX	0710	TID_USTI_ SET_QUEUE_ENTRY_ ERROR	
2	HEX	0711	TID_USTI_ TIMER_INTERVAL_ REQ_FAILED	

Len	Type	Value	Name	Description
2	HEX	0712	TID_USTI_ TIMER_CANCEL_REQ_ FAILED	
2	HEX	0713	TID_USTI_ UTQ_IS_EMPTY	
2	HEX	0801	TID_USDE_ENTRY	
2	HEX	0802	TID_USDE_EXIT	
2	HEX	0803	TID_USDE_RECOVERY	
2	HEX	0804	TID_USDE_ INVALID_FORMAT	
2	HEX	0805	TID_USDE_ INVALID_FUNCTION	
2	HEX	0806	TID_USDE_ DFHUSER_DEQ_FAILED	
2	HEX	0807	TID_USDE_ EXCEPTION_UNKNOWN	
2	HEX	0808	TID_USDE_ LOCK_ERROR	
2	HEX	0809	TID_USDE_ UNLOCK_ERROR	
2	HEX	080A	TID_USDE_ UNLOCK_ERROR_RECOVERY	
	Subpool Names			
8	CHARACTER	USGENRAL	SPNAME_GENERAL	
	Anchor block eyecatcher			
14	CHARACTER	>DFHUSANCHOR	USA_EYE_CATCHER	
	US Lock Name			
8	CHARACTER	USADLOCK	US_ADD_LOCK_NAME	
8	CHARACTER	USXMLOCK	US_TXN_LOCK_NAME	

USGPS User domain statistics

-

CONTROL BLOCK NAME
DFHUSGPC
DESCRIPTIVE NAME
CICS User Domain Statistics
STATUS
LOCATION
The user is passed a pointer to the head of the storage block.

-

User Domain statistics fields.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	28	DFHUSGPS	User domain stats
(0)	UNSIGNED	2	USG_DATA_LENGTH	Length of data area
(2)	UNSIGNED	2	USG_ID	User domain id
(4)	UNSIGNED	1	USG_VERSION	Statistics version number
(5)	CHARACTER	3	*	Reserved
(8)	FULLWORD	4	USG_TIMEOUT_ MEAN_REUSE_TIME	
(C)	FULLWORD	4	USG_TIMEOUT_ REUSE_COUNT	
(10)	FULLWORD	4	USG_TIMEOUT_ EXPIRY_COUNT	
(14)	FULLWORD	4	USG_DIRECTORY_ REUSE_COUNT	
(18)	FULLWORD	4	USG_DIRECTORY_ NOT_FOUND_COUNT	

Constants

Len	Type	Value	Name	Description
1	HEX	01	USG_VERSION_MASK	Version number mask
2	DECIMAL	61	USG_ID_MASK	Stats id mask

USXD User domain transaction data

-

USXD_TRANSACTION_DATA

This structure defines the User-Domain-related transaction storage pointed to by the User Domain transaction token. There is one such structure for every transaction.

It contains one or more user tokens that have been associated with the transaction, together with the pointers to the associated User Data Blocks. One of these pointers is designated as the active UDB pointer, and that is the UDB referenced whenever user attributes for the transaction are queried.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	29	USXD_TRANSACTION_ DATA	
(0)	ADDRESS	4	USXD_ACTIVE	
(4)	ADDRESS	4	USXD_PRINCIPAL	
(8)	ADDRESS	4	USXD_SESSION	
(C)	ADDRESS	4	USXD_EDF	
(10)	ADDRESS	4	USXD_PRINCIPAL_ TOKEN	
(14)	ADDRESS	4	USXD_SESSION_ TOKEN	
(18)	ADDRESS	4	USXD_EDF_TOKEN	
(1C)	BITSTRING	1	USXD_FLAGS	
	1...		USXD_XS_CALLED	XS has been initialized
	.111 1111		*	Reserved

USXT User domain transaction token

-

This structure defines the format of the User Domain transaction token that is preserved by the Transaction Manager. There is one such token for each transaction.

It contains a pointer to the currently active userid for this transaction, and a pointer to the User Domain transaction storage structure.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	USXT_TRANSACTION_ TOKEN	
(0)	ADDRESS	4	USXT_USERID_PTR	Ptr to current userid
(4)	ADDRESS	4	USXT_USXD_PTR	Ptr to transaction data

WBABC Web anchor block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	80	WBAB_WEB_ ANCHOR_BLOCK	
(0)	CHARACTER	16	WBAB_PREFIX	
(0)	HALFWORD	2	WBAB_ANCHOR_ LENGTH	
(2)	CHARACTER	14	WBAB_EYECATCHER	
(10)	ADDRESS	4	*	
(14)	ADDRESS	4	WBAB_DFHWBST_ ENTRY_POINT	
(18)	ADDRESS	4	WBAB_DFHWBTC_ ENTRY_POINT	
(1C)	ADDRESS	4	*	
(20)	ADDRESS	4	WBAB_STATE_ ANCHOR_PTR	
(24)	ADDRESS	4	WBAB_TEMPLATE_ ANCHOR_PTR	
(28)	ADDRESS	4	*	
(2C)	CHARACTER	4	WBAB_3270_ ENVIRONMENT_TOKEN	
(30)	CHARACTER	8	WBAB_STATE_TOKEN	
(38)	CHARACTER	8	WBAB_BUFFER_ TOKEN	
(40)	FULLWORD	4	WBAB_OPENEDITION_ UID	
(44)	ADDRESS	4	WBAB_UNESCAPE_ CODEPAGE_PTR	
(48)	CHARACTER	8	WBAB_MDT_TOKEN	

WBANC Web domain anchor block

-

This anchor block contains the global storage for the WB domain.

It defines the domain state information, variables and constants required by the WB gates.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	72	WBA	

-

Block header

(0)	CHARACTER	16	WBA_PREFIX	===> eyecatcher <===
(0)	HALFWORD	2	WBA_LENGTH	length of wba
(2)	CHARACTER	14	WBA_PREFIX_TEXT	>DFHWBAnchor

--
-

Web Domain state information.

(10)	ADDRESS	4	WBA_LOCK_TOKEN	WB domain lock token
(14)	ADDRESS	4	WBA_STATE_ANCHOR_PTR	
(18)	CHARACTER	8	WBA_GENERAL_ SPTOKEN	token received when subpool was added
(20)	CHARACTER	8	WBA_BUFFER_TOKEN	
(28)	UNSIGNED	1	WBA_WB_STATE	WB domain state initialised, quiesced or terminated
(29)	UNSIGNED	1	WBA_FLAGS	
	1...		WBA_COLD_START	1=CICS cold started
	.1..		WBA_WARM_START	2=CICS warm started
	..11 1111		*	
(2A)	CHARACTER	6	*	padding

--

(30)	ADDRESS	4	WBA_WEBREQUEST_CLASSP	
(34)	ADDRESS	4	WBA_3270_ANCHOR	
(38)	ADDRESS	4	WBA_UNESCAPE_CODEPAGE_PTR	
(3C)	HALFWORD	2	*	Reserved
(3E)	HALFWORD	2	WBA_CODEPAGE_NUMBER	Default codepage number
(40)	CHARACTER	8	WBA_CODEPAGE_ NAME	Default server codepage
(48)	CHARACTER		WBA_END	

--
-

The following is the conversion table for escaped symbols passed to the template manager DFHWBTL. It will contain whatever is coded in DFHCNV for DFHWBUD CLINTCP or, if no DFHWBUD, the default US codepage (see flag bytes to determine which codepage has been used).

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	273	WBA_TTABL	
(0)	CHARACTER	17	WBA_TTABL_HDR	
(0)	HALFWORD	2	WBA_TTABL_LEN	
(2)	CHARACTER	14	WBA_TTABL_ EYECATCH	
(10)	CHARACTER	1	WBA_STARTUP_ FLAGS	
	1111 1...		*	
	.... .1..		WBA_UNESCAPE_TABLE_INITIALIZED	
	.... ..1.		WBA_CCNV_ LOAD_OK	
	1		WBA_WBUD_USED	
(11)	CHARACTER	256	WBA_CONVTABL	each byte addressable

Offset Hex	Type	Len	Name (Dim)	Description
(11)	CHARACTER	1	EBCDIC_VALUE (0 255)	for conversion

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	WB_STATE_ INITIALISING	
1	DECIMAL	2	WB_STATE_ INITIALISED	
1	DECIMAL	3	WB_STATE_QUIESCING	
1	DECIMAL	4	WB_STATE_QUIESCED	
1	DECIMAL	5	WB_STATE_ TERMINATED	

--
-

Literals

Len	Type	Value	Name	Description
8	CHARACTER	WBGENRAL	WB_GENERAL	General purpose subpool for WB domain
14	CHARACTER	>DFHWBANCHOR	WBA_EYE_CATCHER	
8	CHARACTER	WBLOCK	WB_LOCK_NAME	Domain lock
8	CHARACTER	WEBREQAN	WEBREQUEST_ANCHOR	
1	CHARACTER	>	ARROW	
3	CHARACTER	DFH	DFH	

WBA1C Web business logic compatibility interface

-

This copybook defines the 'parameter list' which is passed to program DFHWBA1 to perform the link to the business logic.

A brief description of the fields and their usage follows:

Variable
Type and Usage
wba1_parms_ptr
A pointer variable used as base for the interface parameter list
wba1_parms
top level of the interface parameter list structure
wba1_eyecatcher
A char(8) variable which should contain ' BLIP '
wba1_converter_program_name
A char(8) field containing the name of the program for decode and encode.
wba1_client_address
The IP address of the client.
decode_client_address_string
The IP address of the client in "ww.xx.yy.zz" format.
wba1_data_ptr
A pointer to the storage containing the HTTP request. For BLIO this is an offset.
wba1_method_offset
Offset into the HTTP request of the string containing the method specified for the request.
wba1_http_version_offset
Offset into the HTTP request of the string containing the version for the request.
wba1_resource_offset
Offset into the HTTP request of the string identifying the CICS resource to be invoked for this request.
wba1_header_offset
Offset into the HTTP request of the first HTTP header.
wba1_user_data_offset
Offset into the HTTP request to the "body" of the request - namely any forms data.
wba1_method_length
Length of the string containing the method.
wba1_version_length
Length of the string containing the version of HTTP supported by the client.
wba1_resource_length
Length of the string identifying the CICS resource to be invoked by this HTTP request.
wba1_header_length
Length of the HTTP header request information.(all the headers)
wba1_user_data_length
Length of the HTTP request body.
wba1_input_data_length
Length of the HTTP request body.
wba1_server_program_name
A char(8) name identifying the CICS program that dfhwba1 is to invoke by an EXEC CICS LINK.
wba1_user_token
A fullword token which uniquely identifies the HTTP request being processed.
wba1_outdata_ptr
A pointer to the output data.For BLIO this is an offset.
wba1_response
Response code of this request.
wba1_data
Data for this request if the data is given by offset.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	*	WBA1_PARMS	
(0)	CHARACTER	102	WBA1_PARMS_PLIST	
(0)	CHARACTER	8	WBA1_EYECATCHER	**BLIP** / **BLIO**
(8)	CHARACTER	8	WBA1_CONVERTER_ PROGRAM_NAME	
(10)	UNSIGNED	4	WBA1_CLIENT_ ADDRESS	
(14)	CHARACTER	15	WBA1_CLIENT_ ADDRESS_STRING	
(23)	UNSIGNED	1	WBA1_CLIENT_ ADDRESS_LENGTH	
(24)	FULLWORD	4	*	
(28)	ADDRESS	4	WBA1_DATA_PTR	
(28)	FULLWORD	4	WBA1_DATA_ OFFSET	
(2C)	FULLWORD	4	WBA1_METHOD_ OFFSET	

Offset Hex	Type	Len	Name (Dim)	Description
(30)	FULLWORD	4	WBA1_HTTP_ VERSION_OFFSET	
(34)	FULLWORD	4	WBA1_RESOURCE_ OFFSET	
(38)	FULLWORD	4	WBA1_HEADER_ OFFSET	
(3C)	FULLWORD	4	WBA1_USER_ DATA_OFFSET	
(40)	HALFWORD	2	WBA1_METHOD_ LENGTH	
(42)	HALFWORD	2	WBA1_HTTP_ VERSION_LENGTH	
(44)	HALFWORD	2	WBA1_RESOURCE_ LENGTH	
(46)	HALFWORD	2	WBA1_HEADER_ LENGTH	
(48)	HALFWORD	2	WBA1_USER_ DATA_LENGTH	
(4A)	HALFWORD	2	*	
(4C)	UNSIGNED	4	WBA1_INPUT_ DATA_LENGTH	
(50)	CHARACTER	8	WBA1_SERVER_ PROGRAM_NAME	
(58)	CHARACTER	8	WBA1_USER_TOKEN	
(60)	ADDRESS	4	WBA1_OUTDATA_ PTR	
(60)	FULLWORD	4	WBA1_OUTDATA_ OFFSET	
(64)	UNSIGNED	2	WBA1_RESPONSE	
(66)	CHARACTER	*	WBA1_DATA	

Constants

Len	Type	Value	Name	Description
8	CHARACTER	**BLIP**	WBA1_EYECATCHER_ BLIP	
8	CHARACTER	**BLIO**	WBA1_EYECATCHER_ BLIO	

WBBLC Web business logic interface parameters

-

This copybook defines the 'parameter list' which is passed to program DFHWBBLI to perform the link to the business logic.

A brief description of the fields and their usage follows:

Variable
Type and Usage
wbbl_parms_ptr
A pointer variable used as base for the interface parameter list
wbbl_length
A halfword binary number that must be set to the total length of the BLI parameter list.
wbbl_eyecatcher
A 14-character field that must be set to the standard eyecatcher string '>DFHWBBLIPARMS'.
wbbl_status_size
A one-byte binary field that must be set to the length of the "wbbl_status" substructure (currently 8).
wbbl_mode
A single character that indicates the addressing mode for "wbbl_indata" and "wbbl_outdata". It must be set to 'P' to indicate that these values are pointers, or to 'O' to indicate that these values are offsets (from the start of the parameter list).
wbbl_version
A halfword binary number that indicates which version of the BLI parameter list is currently being used. It should be set using the constant value "wbbl_current_version".
wbbl_prolog_size
A halfword binary number that must be set to the length of the "wbbl_prolog" substructure (currently 56).
wbbl_vector_size
A halfword binary number that must be set to the length of the "wbbl_vector" substructure (currently 64).
wbbl_response
A fullword binary field in which DFHWBBLI returns its response code.
wbbl_client_address
A fullword 32-bit field that must be set to the binary IP address of the client, if this is known.
wbbl_client_address_length
A one-byte binary field that must be set to the length of "wbbl_client_address_string".
wbbl_client_address_string
A string of up to 15 characters which are the "dotted-decimal" representation of "wbbl_client_address", padded on the right with binary zeroes.
wbbl_converter_program_name
The eight-character name of the program that is to be used for converter DECODE and ENCODE functions.
wbbl_server_program_name
The eight-character name of the application program that is to be used to process the request and produce the response.
wbbl_user_token
An eight-character field in which the caller of DFHWBBLI can pass data which identifies the current conversational state with the client. It is usually set to the first eight characters of the +query-string+ portion of the URL (that is, any data following a question mark ('?')).
wbbl_ssl_keysize
Size of the encryption key negotiated during the SSL handshake, if secure sockets layer is being used. Zero if SSL is not being used.
wbbl_indata_ptr
If "wbbl_mode" is 'P', this is the address of the HTTP request data that is to be passed to the application.
wbbl_indata_offset
If "wbbl_mode" is 'O', this field is the offset (from the start of the parameter list) of the HTTP request data that is to be passed to the application.
wbbl_indata_length
A fullword binary number that must be set to the length of the data located by "wbbl_indata_ptr" or "wbbl_indata_offset".

wbbl_outdata_ptr
If "wbbl_mode" is 'P', this is the fullword address in which DFHWBBLI will return the address of the response data from the application. This address is not necessarily the same as "wbbl_indata_ptr".
wbbl_outdata_offset
If "wbbl_mode" is 'O', this is the fullword in which DFHWBBLI will return the offset (from the start of the parameter list) of the response data from the application. This offset is not necessarily the same as "wbbl_indata_offset".
wbbl_outdata_length
The fullword binary field in which DFHWBBLI will return the length of the response data located by "wbbl_outdata_ptr" or "wbbl_outdata_offset".
wbbl_method_offset, wbbl_method_length
Two fullword binary numbers that must contain the offset (from the start of the request data) and the length of the HTTP method that is to be used to process the request. The method should be one of: GET, POST, HEAD, PUT, DELETE, LINK, UNLINK, or REQUEUE.
wbbl_http_version_offset, wbbl_http_version_length
Two fullword binary numbers that must contain the offset (from the start of the request data) and the length of the version of the HTTP protocol that is to be used to process the request.
wbbl_resource_offset, wbbl_resource_length
Two fullword binary numbers that must contain the offset (from the start of the request data) and the length of the URI resource that is being requested (that is, the non-network part of the URL, starting at the first slash ('/') in the URL).
wbbl_header_offset, wbbl_header_length
Two fullword binary numbers that must contain the offset (from the start of the request data) and the length of the HTTP headers associated with this request. This is a list of zero or more headers in the format:

header_name: header_valueCRLF

where the colon and space (': ') delimit the header name from the value, and CRLF (X'0D25') delimits the end of the header value. The end of the list is denoted by an empty header, which contains only a single CRLF.

The first CRLF-delimited line of an HTTP request is not regarded as a header. The offset to the start of the headers is to the character immediately following the CRLF that delimits the first HTTP request line (which may be another CRLF if no headers are present).
wbbl_user_data_offset, wbbl_user_data_length
Two fullword binary numbers that must contain the offset (from the start of the request data) and the length of the body of the HTTP request, if any.
wbbl_client_certificate_offset, wbbl_client_certificate_length
Two fullword binary numbers that must contain the offset (from the start of the request data) and the length of the X.509 client certificate, if any. If the certificate is present, it must be in its binary BER-encoded form, and not base-64 encoded.

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(0)	STRUCTURE	*	WBBL_PARMS	
(0)	CHARACTER	144	WBBL_PARMS_PLIST	
(0)	CHARACTER	16	WBBL_PREFIX	
(0)	HALFWORD	2	WBBL_LENGTH	Length of BLI parmlist
(2)	CHARACTER	14	WBBL_EYECATCHER	
(2)	CHARACTER	1	WBBL_ARROW	Eyecatcher arrow (>)
(3)	CHARACTER	3	WBBL_DFH	Product prefix (DFH)
(6)	CHARACTER	2	WBBL_COMPID	Component id (WB)
(8)	CHARACTER	8	WBBL_BLOCK_ NAME	Block name (BLIPARMS)
(10)	CHARACTER	8	WBBL_STATUS	
(10)	UNSIGNED	1	WBBL_STATUS_ SIZE	Size of this status structure
(11)	CHARACTER	1	WBBL_MODE	'O'=offset, 'P'=pointer
(12)	HALFWORD	2	WBBL_VERSION	Version of WBBL parmlist
(14)	HALFWORD	2	WBBL_PROLOG_ SIZE	Size of WBBL prolog
(16)	HALFWORD	2	WBBL_VECTOR_ SIZE	Size of WBBL vector
(18)	CHARACTER	56	WBBL_PROLOG	
(18)	FULLWORD	4	WBBL_RESPONSE	DFHWBBLI response
(1C)	UNSIGNED	4	WBBL_CLIENT_ADDRESS	Client IP address
(20)	UNSIGNED	1	WBBL_CLIENT_ADDRESS_LENGTH	Length of string
(21)	CHARACTER	15	WBBL_CLIENT_ADDRESS_STRING	

Offset Hex	Type	Len	Name (Dim)	Description
				Dotted-decimal IP ad
(30)	CHARACTER	8	WBBL_CONVERTER_PROGRAM_NAME	Converter program
(38)	CHARACTER	8	WBBL_SERVER_PROGRAM_NAME	Server application
(40)	CHARACTER	8	WBBL_USER_ TOKEN	Token or query string
(48)	UNSIGNED	4	WBBL_SERVER_ADDRESS	Server IP addr
(4C)	HALFWORD	2	WBBL_SERVER_PORTNUMBER	Server portnum
(4E)	HALFWORD	2	WBBL_SSL_ KEYSIZE	SSL key size
(50)	CHARACTER	64	WBBL_VECTOR	
(50)	ADDRESS	4	WBBL_INDATA_ PTR	Addr of request (MODE=P)
(50)	FULLWORD	4	WBBL_INDATA_ OFFSET	Offset of request (MODE=O)
(54)	FULLWORD	4	WBBL_INDATA_ LENGTH	Length of request data
(58)	ADDRESS	4	WBBL_OUTDATA_ PTR	Addr of response (MODE=P)
(58)	FULLWORD	4	WBBL_OUTDATA_OFFSET	Offset to response (MODE=O)
(5C)	FULLWORD	4	WBBL_OUTDATA_LENGTH	Length of response data
(60)	FULLWORD	4	WBBL_METHOD_OFFSET	Offset to request method
(64)	FULLWORD	4	WBBL_METHOD_LENGTH	Length of request method
(68)	FULLWORD	4	WBBL_HTTP_VERSION_OFFSET	Offset to HTTP version
(6C)	FULLWORD	4	WBBL_HTTP_VERSION_LENGTH	Length of HTTP version
(70)	FULLWORD	4	WBBL_RESOURCE_OFFSET	Offset to resource (URL)
(74)	FULLWORD	4	WBBL_RESOURCE_LENGTH	Length of resource
(78)	FULLWORD	4	WBBL_HEADER_ OFFSET	Offset to first HTTP header
(7C)	FULLWORD	4	WBBL_HEADER_LENGTH	Length of all HTTP headers
(80)	FULLWORD	4	WBBL_USER_DATA_OFFSET	Offset to user data (forms)
(84)	FULLWORD	4	WBBL_USER_DATA_LENGTH	Length of user data
(88)	FULLWORD	4	WBBL_CLIENT_CERTIFICATE_ OFFSET	Offset to certificate
(8C)	FULLWORD	4	WBBL_CLIENT_CERTIFICATE_ LENGTH	Length of certificate
(90)	CHARACTER	*	WBBL_DATA	User data (if present)
(90)	CHARACTER	*	WBBL_CLIENT_CERTIFICATE	Certificate data (if present)

Constants

Len	Type	Value	Name	Description
4	DECIMAL	1	WBBL_VERSION_ CTS130	
4	DECIMAL	1	WBBL_CURRENT_ VERSION	
1	CHARACTER	O	WBBL_MODE_OFFSET	
1	CHARACTER	P	WBBL_MODE_POINTER	

WBEPC Web error program parms

-

These declarations define the commarea which is passed
to the user replaceable Web Error program by the CICS WEB
Interface
via a Program Manager Domain EXEC_LINK call.

Variable
Meaning

< wbep_length > (input)
Length of DFHWBEPC copybook

< wbep_eyecatcher >
A character field to contain an eyecatcher
to help with diagnostics.
The caller sets this to '>wbepca '
before calling the Web Error Program

< wbep_version >
Version of DFHWBEPC copybook being passed by CICS

< wbep_error_code > (input)
The two byte signed binary number indicating the cause of the
original error. Constants which this field may contain can be
found in copybook DFHWBUCC.

< wbep_abend_code > (input)
The four character abend code associated with this exception.

< wbep_message_number > (input)
Message number associated with this exception

< wbep_message_ptr > (input)
A pointer to the CICS message text associated with this exception

< wbep_response_len > (input)
The full word length of the HTTP error response to be returned to
the
the HTTP client. On entry to DFHWBEP this contains the default
CICS
HTTP error response for the reported error.

< wbep_response_ptr > (input)
A pointer to the 32K buffer containing the HTTP error response to
be
returned to the HTTP client. On entry to DFHWBEP this contains
the
default HTTP error response returned by CICS for the reported
error.

< wbep_response_len > (input)
The full word length of the response message text associated with
this exception.

< wbep_client_address_len > (input)
One byte field containing the length of the address contained in
wbep_client_address

< wbep_client_address > (input)
The 15 character TCPIP address of the client.

< wbep_server_address_len > (input)
One byte field containing the length of the address contained in
wbep_server_address

< wbep_server_address > (input)
The 15 character TCPIP address of the TCP/IP stack on which this
request was received

< wbep_tcpipservice_name > (input)
Name of the TCPIPSERVICE associated with the failing request

< wbep_converter_program < (input)
The 8 character name of the converter program associated with
this
request

< wbep_target_program > (input)
The target program associated with the web request.

< wbep_failing_program > (input)
The program which CICS was invoking when the failure occurred

< wbep_http_response_code > (input)
HTTP error response code CICS is returning for this error.
This can be overriden by changing the content of the buffer
containing the HTTP response

< wbep_analyzer_response > (input)
Response code returned by analyzer program

< wbep_analyzer_reason > (input)
Reason code returned by analyzer program

< wbep_converter_response > (input)
Response code returned by converter program

< wbep_converter_reason > (input)
Reason code returned by converter program

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	204	DFHWBEPC	
(0)	CHARACTER	12	WBEP_PREFIX	
(0)	HALFWORD	2	WBEP_LENGTH	
(2)	CHARACTER	8	WBEP_EYECATCHER	
(A)	HALFWORD	2	WBEP_VERSION	
(C)	CHARACTER	112	WBEP_DATA	
(C)	HALFWORD	2	WBEP_ERROR_CODE	
(E)	CHARACTER	2	*	
(10)	CHARACTER	4	WBEP_ABEND_CODE	
(14)	FULLWORD	4	WBEP_MESSAGE_NUMBER	
(18)	ADDRESS	4	WBEP_MESSAGE_ PTR	
(1C)	FULLWORD	4	WBEP_MESSAGE_ LEN	
(20)	ADDRESS	4	WBEP_RESPONSE_ PTR	
(24)	FULLWORD	4	WBEP_RESPONSE_ LEN	
(28)	UNSIGNED	1	WBEP_CLIENT_ADDRESS_LEN	
(29)	CHARACTER	15	WBEP_CLIENT_ ADDRESS	
(38)	UNSIGNED	1	WBEP_SERVER_ADDRESS_LEN	
(39)	CHARACTER	15	WBEP_SERVER_ADDRESS	
(48)	CHARACTER	8	WBEP_TCPIPSERVICE_NAME	
(50)	CHARACTER	8	WBEP_CONVERTER_PROGRAM	
(58)	CHARACTER	8	WBEP_TARGET_PROGRAM	
(60)	CHARACTER	8	WBEP_FAILING_PROGRAM	
(68)	FULLWORD	4	WBEP_HTTP_RESPONSE_CODE	
(6C)	FULLWORD	4	WBEP_ANALYZER_RESPONSE	
(70)	FULLWORD	4	WBEP_ANALYZER_REASON	
(74)	FULLWORD	4	WBEP_CONVERTER_RESPONSE	
(78)	FULLWORD	4	WBEP_CONVERTER_REASON	
(7C)	CHARACTER	80	*	

WBSTC Web state manager data

-

This file contains state data structure and the state anchor block declarations.

-

wbsth_prefix Eyecatcher for state block
wbsth_partnership_status The state of the task relationship
wbsth_master_taskid Task number of master transaction
wbsth_master_cuowid CICS uow id for master transaction
wbsth_master_ecb ECB for master transaction
wbsth_slave_taskid Task number of slave transaction
wbsth_slave_cuowid CICS uow id for slave transaction
wbsth_slave_ecb ECB for slave transaction
wbsth_timestamp Timestamp of this state block
wbsth_user_data The state user data

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	272	WBSTH_STATE_BLOCK	
(0)	CHARACTER	16	WBSTH_PREFIX	
(0)	HALFWORD	2	WBSTH_PREFIX_ LENGTH	
(2)	CHARACTER	14	WBSTH_PREFIX_ TEXT	
(10)	UNSIGNED	4	WBSTH_PARTNERSHIP_ STATUS	
(14)	CHARACTER	4	WBSTH_MASTER_ TASKID	
(18)	CHARACTER	8	WBSTH_MASTER_ CUOWID	
(20)	UNSIGNED	4	WBSTH_MASTER_ECB	
(20)	UNSIGNED	1	*	
(21)	UNSIGNED	3	WBSTH_M_C_CODE	
(24)	CHARACTER	4	WBSTH_SLAVE_ TASKID	
(28)	CHARACTER	8	WBSTH_SLAVE_ CUOWID	
(30)	UNSIGNED	4	WBSTH_SLAVE_ECB	
(30)	UNSIGNED	1	*	
(31)	UNSIGNED	3	WBSTH_S_C_CODE	
(34)	UNSIGNED	4	WBSTH_TIMESTAMP	
(38)	CHARACTER	216	WBSTH_USER_DATA	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	216	WBSTU_STATE_DATA	
(0)	CHARACTER	8	WBSTU_FACILITY_ TOKEN	
(8)	CHARACTER	4	WBSTU_TARGET_ TRANSACTION_ID	
(C)	CHARACTER	4	WBSTU_NEXT_ TRANSACTION_ID	
(10)	CHARACTER	4	WBSTU_TERMID	
(14)	CHARACTER	4	WBSTU_TARGET_ ABEND_CODE	
(18)	CHARACTER	8	WBSTU_TCPIPSERVICE	
(20)	CHARACTER	8	WBSTU_BMS_ PAGE_TOKEN	
(28)	ADDRESS	4	WBSTU_3270_ PAGE_TOKEN	
(2C)	ADDRESS	4	WBSTU_MDT_ TABLE_PTR	
(30)	ADDRESS	4	WBSTU_OUTPUT_ DATA_PTR	
(34)	FULLWORD	4	WBSTU_OUTPUT_ DATA_LENGTH	
(38)	ADDRESS	4	WBSTU_OUTPUT_ OFFSET	
(3C)	ADDRESS	4	WBSTU_OUTPUT_ LENGTH_REMAINING	
(40)	ADDRESS	4	WBSTU_INPUT_ DATA_PTR	
(44)	FULLWORD	4	WBSTU_INPUT_ DATA_LENGTH	
(48)	CHARACTER	8	WBSTU_EXPORTED_ DOCUMENT	
(48)	ADDRESS	4	WBSTU_EXPORTED_ DOCUMENT_PTR	
(4C)	FULLWORD	4	WBSTU_EXPORTED_ DOCUMENT_LEN	
(50)	UNSIGNED	1	WBSTU_CONVERSATION_ TYPE	
(51)	UNSIGNED	1	WBSTU_AID	
(52)	HALFWORD	2	WBSTU_CURSOR	

Offset Hex	Type	Len	Name (Dim)	Description
(54)	BITSTRING	1	WBSTU_USER_STATE	
	1...		WBSTU_PSEUDO_ CONVERSATION	
	.1..		WBSTU_DATA_TYPE	
	..1.		WBSTU_INITIAL_ RECEIVE	
	...1		WBSTU_LAST_ SEND_WSF_QUERY	BA17417A
	 1111		*	BA17417C
(55)	UNSIGNED	1	* (2)	
(57)	UNSIGNED	1	WBSTU_URL_LENGTH	
(58)	CHARACTER	128	WBSTU_URL	

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	36	WBSTA_ANCHOR_ BLOCK	
(0)	CHARACTER	16	WBSTA_ANCHOR_ PREFIX	
(0)	HALFWORD	2	WBSTA_ANCHOR_ PREFIX_LEN	
(2)	CHARACTER	14	WBSTA_ANCHOR_ PREFIX_TEXT	
(10)	UNSIGNED	4	WBSTA_GARBAGE_ INTERVAL	In minutes
(14)	CHARACTER	4	WBSTA_DIRECTORY_ TOKEN	
(18)	ADDRESS	4	WBSTA_LOCK_TOKEN	
(1C)	CHARACTER	4	WBSTA_WAKEUP_ TIME	As 0hhmmssC
(20)	UNSIGNED	4	WBSTA_TERMINAL_ TIMEOUT	In minutes

Constants

Len	Type	Value	Name	Description
1	DECIMAL	0	WBSTH_NOT_ INITIALIZED	
1	DECIMAL	1	WBSTH_INITIALIZED	
1	DECIMAL	2	WBSTH_MADE	
1	DECIMAL	3	WBSTH_BROKEN	
1	DECIMAL	4	WBSTH_TERMINATED	
1	DECIMAL	0	WBSTU_NEW_ CONVERSATION	
1	DECIMAL	1	WBSTU_MAP_ CONVERSATION	
1	DECIMAL	2	WBSTU_TEXT_ CONVERSATION	
1	DECIMAL	3	WBSTU_TC_ CONVERSATION	

WBUCC Web interface urp constants

```
-


 This copybook defines the constants which are used by
 the User Replaceable Programs.

 < Constant >
 Meaning

 < URP_DECODE >
 The call is to the decode function of the converter program.

 < URP_ENCODE >
 The call is to the encode function of the converter program.

 < URP_OK >
 The RESPONSE value from the User Replaceable Program is OK.

 < URP_EXCEPTION >
 The RESPONSE value from the User Replaceable Program is
 EXCEPTION.

 < URP_INVALID >
 The RESPONSE value from the User Replaceable Program is
 INVALID.

 < URP_DISASTER >
 The RESPONSE value from the User Replaceable Program is
 DISASTER.

 < URP_CORRUPT_CLIENT_DATA >
 An architected
 REASON for an EXCEPTION response produced by the
 converter decode function.

 < URP_SECURITY_FAILURE >
 An architected
 REASON for an EXCEPTION response produced by the
 converter decode function.

 < URP_RESOURCE_TOO_SHORT >
 Reason code returned by CICS-supplied default Analyzer DFHWBADX
 if
 the URI on the HTTP Request is shorter than that expected by the
 default analyzer.

 < URP_FIRST_SLASH_MISSING >
 Reason code returned by CICS-supplied default Analyzer DFHWBADX
 if
 it cannot locate a an EBCDIC "/" character in the URI of the
 incoming
 data.

 < URP_CONV_NAME_INVALID >
 Reason code returned by CICS-supplied default Analyzer DFHWBADX
 if
 it detects that the name of the converter program to be invoked
 for
 this request is greater than 8 bytes long or has a length of
 zero.

 < URP_TRAN_NAME_INVALID >
 Reason code returned by CICS-supplied default Analyzer DFHWBADX
 if
 it detects that the name of the transaction to be started by CICS
 to
 process this request is greater than 8 bytes long or has a
 length of zero.

 < URP_SERV_NAME_INVALID >
 Reason code returned by CICS-supplied default Analyzer DFHWBADX
 if
 it detects that the name of the target program to be invoked for
 this request is greater than 8 bytes long, or has a length of
 zero.
```

< URP_USER_TOKEN_INVALID >
Reason code returned by CICS-supplied default Analyzer DFHWBADX
if
it detects that the name of the target program to be invoked for
this request is greater than 8 bytes long, or has a length of
zero.

< URP_SERVER_NAME_MISSING >
Reason code returned by CICS-supplied default Analyzer DFHWBADX
if
it cannot identify the name of the target program from the URI in
the
HTTP request received.

< eyecatchers >
Definitions of the eyecatchers at the front
of the COMMAREAs passed to the Web Interface
user replaceable programs.

Converter Function Types

Offset Hex	Type	Len	Name (Dim)	Description
2	DECIMAL	1	URP_DECODE	
2	DECIMAL	2	URP_ENCODE	
URP Response Values				
4	DECIMAL	0	URP_OK	
4	DECIMAL	4	URP_EXCEPTION	
4	DECIMAL	8	URP_INVALID	
4	DECIMAL	12	URP_DISASTER	
URP: Converter reasons for exception response				
4	DECIMAL	1	URP_SECURITY_ FAILURE	
4	DECIMAL	2	URP_CORRUPT_CLIENT_DATA	
URP: Analyzer reasons for exception response				
4	DECIMAL	1	URP_RESOURCE_TOO_SHORT	
4	DECIMAL	2	URP_FIRST_SLASH_MISSING	
4	DECIMAL	4	URP_CONV_ NAME_INVALID	
4	DECIMAL	5	URP_TRAN_ NAME_INVALID	
4	DECIMAL	6	URP_SERV_ NAME_INVALID	
4	DECIMAL	7	URP_USER_TOKEN_INVALID	
4	DECIMAL	8	URP_SERVER_NAME_MISSING	
Eyecatcher values				
8	CHARACTER	>decode	DECODE_EYECATCHER_INIT	
8	CHARACTER	>encode	ENCODE_EYECATCHER_INIT	
8	CHARACTER	>analyze	ANALYZE_EYECATCHER_INIT	
8	CHARACTER	>dfhwbun	DFHWBUN_EYECATCHER_INIT	
DFHWBUN current version				
4	DECIMAL	1	DFHWBUN_CURRENT_VERSION	
DFHCNV keys				
8	CHARACTER	DFHWBHH	CONV_HTTP_ HEADER_KEY	
8	CHARACTER	DFHWBUD	CONV_USER_DATA_KEY	
Possible values of wbra_ request_type				
4	DECIMAL	1	WBRA_TYPE_HTTP	
4	DECIMAL	2	WBRA_TYPE_NON_HTTP	
4	DECIMAL	3	WBRA_UNESCAPE_REQUIRED	
4	DECIMAL	4	WBRA_UNESCAPE_NOT_REQUIRED	
Possible values of wbep_ error_code				
2	DECIMAL	1	WBEP_BLIO_GREATER_THAN_32K_RESPONSE	
2	DECIMAL	2	WBEP_COMMAREA_NO_CONTENT	
2	DECIMAL	3	WBEP_DFHWBBLI_DOCUMENT_NOT_FOUND	

Offset Hex	Type	Len	Name (Dim)	Description
2	DECIMAL	4	WBEP_DFHWBBLI_ CODEPAGE_NOT_FOUND	
2	DECIMAL	5	WBEP_DFHWBBLI_ API_ERROR	
2	DECIMAL	6	WBEP_DFHWBBLI_ LINK_FAILED_TERMERR	
2	DECIMAL	7	WBEP_DFHWBBLI_ LINK_FAILED_INVREQ	
2	DECIMAL	8	WBEP_DFHWBBLI_ LINK_FAILED_LENGERR	
2	DECIMAL	9	WBEP_DFHWBBLI_ LINK_FAILED_PGMIDERR	
2	DECIMAL	10	WBEP_DFHWBBLI_ LINK_FAILED_SYSIDERR	
2	DECIMAL	11	WBEP_DFHWBBLI_ LINK_FAILED_ROLLEDBACK	
2	DECIMAL	12	WBEP_DFHWBBLI_ LINK_FAILED_NOTAUTH	
2	DECIMAL	13	WBEP_DFHWBBLI_ LINK_FAILED	
2	DECIMAL	14	WBEP_INVALID_ DECODE_PARAMETER_ LIST	
2	DECIMAL	15	WBEP_DECODE_ERROR	
2	DECIMAL	16	WBEP_INVALID_ ENCODE_PARAMETER_ LIST	
2	DECIMAL	17	WBEP_ENCODE_ERROR	
2	DECIMAL	18	WBEP_SAVE_ CERTIFICATE_FAILED	
2	DECIMAL	19	WBEP_DFHWBBLI_ ABEND_HANDLER_ INVOKED	
2	DECIMAL	20	WBEP_INVALID_ ATTACH	
2	DECIMAL	21	WBEP_RECEIVE_ERROR	
2	DECIMAL	22	WBEP_ANALYZER_ LINK_ERROR	
2	DECIMAL	23	WBEP_DFHWBXN_ CODEPAGE_ERROR	
2	DECIMAL	24	WBEP_NO_ANALYZER_ SPECIFIED	
2	DECIMAL	25	WBEP_RECEIVE_ STORAGE_ERROR	
2	DECIMAL	26	WBEP_HEADER_ LENGTH_ERROR	
2	DECIMAL	27	WBEP_DFHWBXN_ LOGIC_ERROR	
2	DECIMAL	28	WBEP_LINK_ DFHWBBLI_FAILED	
2	DECIMAL	29	WBEP_ANALYZER_ ERROR	
2	DECIMAL	30	WBEP_ANALYZER_ DATALENGTH_ERROR	
2	DECIMAL	31	WBEP_NOT_ AUTHORIZED_TO_ START_ALIAS	
2	DECIMAL	32	WBEP_DFHWBBLI_ BAD_PREVIOUS_WEB_ SEND	
2	DECIMAL	33	WBEP_BAD_ COMMAREA_RESPONSE	

WRB Web request block class

-

this copybook encapsulates the code and control blocks associated with the processing of an HTTP (or non-HTTP) request received on a port associated with a CICS Web TCPIPSERVICE.

Each request is represented by a WebRequest object (wrb). The WebRequests form a doubly-linked list which is anchored in the Web anchor block (wba). The WebRequest object contains all the information needed to process the request.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	DeclareClass	4	WEBREQ	
INSTANCE DATA				
Declared Data				
(0)	CHAR Private	4	*	

-

The following control blocks are defined:

WebRequest class anchor block (wra)
contains class related private information, including the anchor for the chain of class objects currently installed. Created during initialization of the Web Domain. Lives for the lifetime of CICS.
WebRequest class object (wrb)
Contains information about a Class object which is currently installed - created when incoming data arrives on a Port with CWXN specified as the transaction to be started to process the new work. Chained together as a linked list.
WebRequest class browse block (wrbr)
Contains information about an ongoing browse of the WebRequest objects. Created at INQUIRE START, and destroyed at INQUIRE END.

WRA - WebRequest class anchor block

Offset Hex	Type	Len	Name (Dim)	Description
SHARED DATA				
Declared Data				
(0)	CHAR Protected	48	WRA	
(0)	CHAR Protected	16	WRA_PREFIX	
(0)	SIGNED Protected	2	WRA_LENGTH	length of wra
(2)	CHAR Protected	1	WRA_ARROW	
(3)	CHAR Protected	3	WRA_DFH	
(6)	CHAR Protected	2	WRA_DOMID	
(8)	CHAR Protected	8	WRA_BLOCK_NAME	
(10)	CHAR Protected	8	WRA_WRB_SPTOKEN	wrb subpool token
(18)	CHAR Protected	8	WRA_WRBR_ SPTOKEN	wrbr subpool token
(20)	CHAR Protected	8	*	
(20)	ADDRESS Protected	4	WRA_WRB_FIRST	-> first wrb
(24)	ADDRESS Protected	4	WRA_WRB_LAST	-> last wrb
(28)	CHAR Protected	8	WRA_WRBRHEAD	
(28)	ADDRESS Protected	4	WRA_WRBR_FIRST	-> first tbr
(2C)	ADDRESS Protected	4	WRA_WRBR_LAST	-> last tbr
(30)	CHAR Protected		*	

Header for wrb chain.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Protected	*	WRA_WRBHEAD	

WRB - WebRequest

Offset Hex	Type	Len	Name (Dim)	Description
(0)	CHAR Public	392	WRB	
(0)	CHAR Public	16	WRB_PREFIX	
(0)	SIGNED Public	2	WRB_LENGTH	WRB control block length
(2)	CHAR Public	14	WRB_EYECATCHER	Eyecatcher '>DFHWBREQBLK'
(10)	ADDRESS Public	4	WRB_NEXT	-> next wrb
(14)	ADDRESS Public	4	WRB_PREV	-> previous wrb
(18)	BITSTRING Public	1	WRB_FLAGS1	
	`1... .... Protected`		WRB_GREATER_THAN_32K	

Offset Hex	Type	Len	Name (Dim)	Description
	.1.. .... Protected		WRB_FIRST_LINE_COMPLETE	
	..1. .... Protected		WRB_SHARED_TS_REPOSITORY	
	...1 .... Protected		WRB_RECEIVE_COMPLETE	
	.... 1... Protected		WRB_HEADERS_RECEIVED	
	.... .1.. Protected		WRB_INITIAL_ BUFFER	
	.... ..1. Protected		WRB_EXEC_CICS_WEB_SEND	
	....1 Protected		WRB_SEND_ DOCUMENT	
(19)	BITSTRING Public	1	WRB_FLAGS2	
	1... .... Protected		WRB_CONNECTION_PERSISTENT	
	.1.. .... Protected		WRB_CONTENT_LENGTH_FOUND	
	..1. .... Protected		WRB_CONTENT_LENGTH_SENT	
	...1 .... Protected		WRB_KEEP_ALIVE_SENT	
	.... 1... Protected		WRB_USER_DATA_ESCAPED	
	.... .1.. Protected		WRB_FIRST_RECV_IN_REQUEST	
	.... ..1. Protected		WRB_TIDYUP_COMPLETE	
	....1 Protected		WRB_SEND_RESPONSE_FAILED	
(1A)	CHAR Public	2	*	
(1C)	CHAR Public	8	WRB_SESSION_ TOKEN	
(1C)	ADDRESS Public	4	WRB_SESSION_TOKEN_PART1	
(20)	UNSIGNED Public	4	WRB_SESSION_TOKEN_PART2	
(24)	UNSIGNED Public	1	WRB_REQUEST_ TYPE	
(25)	UNSIGNED Public	1	*	
(26)	SIGNED Public	2	WRB_KEYSIZE	
(28)	CHAR Public	8	WRB_USERID	
(30)	ADDRESS Public	4	WRB_SERVER_DATA_PTR	
(34)	SIGNED Public	4	WRB_REMAINING_BUFFER_LEN	
(38)	CHAR Public	8	WRB_SERVER_PROGRAM_NAME	
(40)	CHAR Public	8	WRB_CONVERTER_PROGRAM_NAME	
(48)	CHAR Public	8	WRB_USER_TOKEN	
(50)	UNSIGNED Public	4	WRB_CLIENT_ ADDRESS	
(54)	UNSIGNED Public	4	WRB_SERVER_ ADDRESS	
(58)	CHAR Public	16	WRB_CHAR_ CLIENT_ADDRESS_AREA	
(58)	UNSIGNED Public	1	WRB_CHAR_ CLIENT_ADDRESS_LEN	
(59)	CHAR Public	15	WRB_CHAR_CLIENT_ADDRESS	
(68)	CHAR Public	16	WRB_CHAR_ SERVER_ADDRESS_AREA	
(68)	UNSIGNED Public	1	WRB_CHAR_ SERVER_ADDRESS_LEN	
(69)	CHAR Public	15	WRB_CHAR_SERVER_ADDRESS	
(78)	CHAR Public	44	WRB_COMMON	
(78)	SIGNED Public	4	WRB_METHOD_ OFFSET	
(7C)	SIGNED Public	4	WRB_METHOD_ LENGTH	
(80)	SIGNED Public	4	WRB_RESOURCE_OFFSET	
(84)	SIGNED Public	4	WRB_RESOURCE_LENGTH	
(88)	SIGNED Public	4	WRB_HTTP_VERSION_OFFSET	
(8C)	SIGNED Public	4	WRB_HTTP_VERSION_LENGTH	
(90)	SIGNED Public	4	WRB_HEADER_ OFFSET	
(94)	SIGNED Public	4	WRB_HEADER_ LENGTH	
(98)	SIGNED Public	4	WRB_USER_DATA_OFFSET	
(9C)	SIGNED Public	4	WRB_USER_DATA_LENGTH	
(A0)	SIGNED Public	4	*	

Offset Hex	Type	Len	Name (Dim)	Description
(A4)	UNSIGNED Public	4	WRB_INPUT_DATA_LENGTH	
(A8)	UNSIGNED Public	4	WRB_RECEIVE_BUFFER_OFFSET	
(AC)	UNSIGNED Public	2	WRB_HEADER_ NUMBER	
(AE)	UNSIGNED Public	2	WRB_USER_NUMBER	
(B0)	UNSIGNED Public	4	WRB_BYTES_ RECEIVED	
(B4)	UNSIGNED Public	4	WRB_CONTENT_ LENGTH	
(B8)	ADDRESS Public	4	WRB_CURRENT_PTR	
(BC)	ADDRESS Public	4	WRB_OUTDATA_PTR	
(C0)	UNSIGNED Public	4	WRB_OUTDATA_ LENGTH	
(C4)	CHAR Public	8	WRB_DFHCNV_KEY	
(CC)	CHAR Public	8	WRB_SERVER_PROTOCOL	
(D4)	CHAR Public	4	WRB_TASK_NUM	
(D8)	CHAR Public	4	WRB_REPOSITORY_ STCK	
(DC)	UNSIGNED Public	4	WRB_FIRST_LINE_LENGTH	
(E0)	CHAR Public	8	WRB_ANALYZER_ NAME	
(E8)	SIGNED Public	4	WRB_ANALYZER_RESPONSE	
(EC)	SIGNED Public	4	WRB_ANALYZER_REASON	
(F0)	SIGNED Public	4	WRB_CONVERTER_RESPONSE	
(F4)	SIGNED Public	4	WRB_CONVERTER_REASON	
(F8)	ADDRESS Public	4	WRB_HEADER_BROWSE_TOKEN	
(FC)	SIGNED Public	4	WRB_HEADER_BROWSE_OFFSET	
(100)	SIGNED Public	4	WRB_USER_DATA_CURSOR	
(104)	SIGNED Public	4	WRB_RESPONSE_HEADER_LEN	
(108)	CHAR Public	8	WRB_REPOSITORY_TOKEN	
(110)	CHAR Public	6	WRB_REPOSITORY_HEADER	
(116)	SIGNED Public	2	WRB_SERVER_PORTNUMBER	
(118)	CHAR Public	8	WRB_CERT_REPOSITORY_TOKEN	
(120)	CHAR Public	40	WRB_CLIENT_ CODEPAGE	
(148)	CHAR Public	8	WRB_TCPIPSERVICE	
(150)	ADDRESS Public	4	WRB_RECEIVE_DATA_PTR	
(154)	ADDRESS Public	4	WRB_OVERLEN_DATA_PTR	
(158)	CHAR Public	16	WRB_NEW_SEND_DOCTOKEN	
(168)	SIGNED Public	4	WRB_RESPONSE_LINE_LENGTH	
(16C)	SIGNED Public	4	WRB_SEND_BODY_LENGTH	
(170)	CHAR Public	8	WRB_FAILING_ PROGRAM	
(178)	CHAR Public	8	WRB_INITIAL_ STRING	
(180)	CHAR Public	4	WRB_ABEND_CODE	
(184)	SIGNED Public	2	WRB_ERROR_CODE	
(186)	CHAR Public	2	*	
WRBR - WebRequest browse block				
(0)	CHAR Protected	40	WRBR	
(0)	ADDRESS Protected	4	WRBR_NEXT	-> next wrbr
(4)	ADDRESS Protected	4	WRBR_PREV	-> previous wrbr
(8)	CHAR Protected	4	WRBR_TRANID	browsing tranid
(C)	CHAR Protected	4	WRBR_TRANNUM	browsing tran number
(10)	CHAR Protected	8	WRBR_TRANTOKEN	browsing tran token
(18)	CHAR Protected	4	WRBR_TOKEN	cursor value
(1C)	SIGNED Protected	4	WRBR_CHANGE_ COUNT	change count at last get_next
(20)	ADDRESS Protected	4	WRBR_WRBP	-> current wrbr
(24)	ADDRESS Protected	4	*	reserved

--

Offset Hex	Type	Len	Name (Dim)	Description
(0)	FIXED Public	1	TRUNCATE	
(0)	FIXED Public	1	SET	
(0)	FIXED Public	1	PERSIST	
(0)	FIXED Public	1	INITIAL	
(0)	FIXED Public	1	CONVERT	
(0)	FIXED Public	4	WRQ_RESPONSE	

Constants

Len	Type	Value	Name	Description
4	DECIMAL	392	WRB_ROUNDED_ UP_LENGTH	
1	DECIMAL	0	TRUNCATE_NO	
1	DECIMAL	1	TRUNCATE_YES	
1	DECIMAL	0	SET_NO	
1	DECIMAL	1	SET_YES	
1	DECIMAL	0	PERSIST_NO	
1	DECIMAL	1	PERSIST_YES	
1	DECIMAL	0	INITIAL_NO	
1	DECIMAL	1	INITIAL_YES	
1	DECIMAL	0	CONVERT_NO	
1	DECIMAL	1	CONVERT_YES	
4	DECIMAL	1	WRQ_OK	
4	DECIMAL	3	WRQ_PURGED	
4	DECIMAL	4	WRQ_DISASTER	
4	DECIMAL	5	WRQ_SOCKETS_ RECEIVE_ERROR	
4	DECIMAL	6	WRQ_STORAGE_ERROR	
4	DECIMAL	7	WRQ_ANALYZER_ LINK_ERROR	
4	DECIMAL	8	WRQ_ANALYZER_ERROR	
4	DECIMAL	9	WRQ_SOCKETS_ SEND_ERROR	
4	DECIMAL	10	WRQ_SOCKETS_ CLOSE_ERROR	
4	DECIMAL	11	WRQ_SOIS_ INQUIRE_FAILED	
4	DECIMAL	12	WRQ_NO_ANALYZER	
4	DECIMAL	13	WRQ_NOT_HTTP_ REQUEST	
4	DECIMAL	14	WRQ_WBQM_ PUT_HEADER_FAILED	
4	DECIMAL	15	WRQ_WBQM_ PUT_USER_FAILED	
4	DECIMAL	16	WRQ_NOT_WEB_ REQUEST	
4	DECIMAL	17	WRQ_HDR_BROWSE_ ACTIVE	
4	DECIMAL	18	WRQ_HDR_BROWSE_ NOT_ACTIVE	
4	DECIMAL	19	WRQ_REPOSITORY_ IO_ERROR	
4	DECIMAL	20	WRQ_HDR_BROWSE_END	
4	DECIMAL	21	WRQ_HDR_NOT_FOUND	
4	DECIMAL	22	WRQ_INVALID_ REQUEST_FORMAT	
4	DECIMAL	23	WRQ_HDR_VALUE_ LENGTH_ERROR	
4	DECIMAL	24	WRQ_HDR_NAME_ LENGTH_ERROR	
4	DECIMAL	25	WRQ_INVALID_HEADER	
4	DECIMAL	26	WRQ_DOCUMENT_ NOT_FOUND	
4	DECIMAL	27	WRQ_CODEPAGE_ NOT_FOUND	
4	DECIMAL	28	WRQ_WBQM_ GET_REPTOKEN_ERR	
4	DECIMAL	29	WRQ_WBQM_ GET_BODY_OUT_FAILED	
4	DECIMAL	30	WRQ_WBQM_ GET_RESPLINE_FAILED	
4	DECIMAL	31	WRQ_WBQM_ GET_HEADER_OUT_ FAILED	
4	DECIMAL	32	WRQ_CONNECTION_ CLOSED	
4	DECIMAL	33	WRQ_HDR_LENGTH_ ERROR	
4	DECIMAL	34	WRQ_ANALYZER_ DATALENG_ERROR	
4	DECIMAL	35	WRQ_NO_PREVIOUS_ SEND	
4	DECIMAL	36	WRQ_BAD_PREVIOUS_ SEND	

XCCBC External CICS interface control blocks

```
CONTROL BLOCK NAME = DFHXCCBC
DESCRIPTIVE NAME = CICS External CICS Interface Control
                Block definitions
FUNCTION =
    This file contains the control block and constant
    declarations used by the External CICS Interface.
    The file is included in each EXCI module.
    The control blocks are:
       XCGLOBAL - XCGLOBAL block
       XCUSER - XCUSER block
       XCPIPE - XCPIPE block
    All blocks are MVS GETMAINED from storage above the 16MB
    line, subpool 1.
LIFETIME =
    There is only ever one XCGLOBAL block per TCB, and it is
    created on the first Initialise_user call for that TCB. It
    remains until TCB Termination.
    An XCUSER Block is created for each new 'user' defined to
    the system via an Initialise_user call. It remains until
    TCB termination.
    An XCPIPE block is created when an allocate_pipe EXCI
    request is issued for a particular user. It is destroyed
    when a deallocate_pipe request is issued, or at TCB
    termination.
LOCATION =
    XCGLOBAL is chained off the batch AFCB.
    XCUSER blocks are chained together and anchored off XCGLOBAL
    XCPIPE blocks for a particular user are chained together
    and anchored off the relevant XCUSER.
NOTES :
 DEPENDENCIES = S/370
 RESTRICTIONS = none
 MODULE TYPE = Control block definition
  XCGGLOBAL Block
```

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	337	XCGLOBAL	
(0)	CHARACTER	16	XCG_PREFIX	Standard Prefix
(0)	HALFWORD	2	XCG_LENGTH	
(2)	CHARACTER	14	XCG_EYE	>XC_GLOBAL
Module addresses				
(10)	ADDRESS	4	XCG_PRH_ADDR	Entry Point of DFHXCPRH
(14)	ADDRESS	4	XCG_XFQ_ADDR	Entry Point of DFHXFQ
(18)	ADDRESS	4	XCG_EIP_ADDR	Entry Point of DFHXCEIP
(1C)	ADDRESS	4	XCG_TRP_ADDR	Entry Point of DFHXCTRP
(20)	ADDRESS	4	XCG_TRI_ADDR	Entry Point of DFHXCTRI
(24)	ADDRESS	4	XCG_DMP_ADDR	Entry Point of DFHXCDMP
(28)	ADDRESS	4	XCG_URM_ADDR	Entry Point of DFHXCURM
(2C)	ADDRESS	4	XCG_TRA_ADDR	Entry Point of DFHXCTRA
(30)	ADDRESS	4	XCG_MSG_ADDR	Entry Point of DFHMEBM
(34)	ADDRESS	4	XCG_MTAB_ADDR	Entry Point of DFHMET4E
Working Storage addresses. For XCEIP there is only ever one instance of EIP's working storage, as all EXEC requests are funelled through one user called DFHXCEIP. For XCPRH, XCG_PRH_WS points to the working storage of DFHXCPRH for the currently active user. Each user will have its XCPRH's working storage hung of its XCUSER block.				
(38)	ADDRESS	4	XCG_PRH_WS	Addr(DFHXCPRH's working stg)
(3C)	ADDRESS	4	XCG_EIP_WS	Addr(DFHXCEIP's working stg)
(40)	FULLWORD	4	XCG_PRH_WS_LEN	Len(DFHXCPRH's working stg)
(44)	FULLWORD	4	XCG_EIP_WS_LEN	Len(DFHXCEIP's working stg)
URM Global fields .				
(48)	ADDRESS	4	XCG_URM_ANCHOR	URM global storage anchor
(4C)	CHARACTER	8	XCG_PROGRAM	Server program name
Parameters for Trace and message facilities				
(54)	ADDRESS	4	XCG_TRAP_WA_PTR	DFHXCTRA's work area address
(58)	ADDRESS	4	XCG_TRACE_ANCHOR	Trace anchor block address
(5C)	UNSIGNED	4	XCG_TRACE_ TABLE_SIZE	Trace table size
(60)	CHARACTER	1	XCG_TRACE_LVL	Level of tracing required
	1...		LEVEL1	Tracing level 1 required
	.1..		LEVEL2	Tracing level 2 required
	..11 1111		*	Reserved
(61)	BITSTRING	1	XCG_TRACE_FLAGS	Trace flags
	1...		XCG_GTF_STARTED	Initial GTF status

Offset Hex	Type	Len	Name (Dim)	Description
	.1..		XCG_TRAP_ACTIVE	Initial status of TRAP
	..1.		XCG_TRACE_ CONFDATA	CONFDATA=HIDETC
	...1 1111		*	Reserved
(62)	BITSTRING	1	XCG_MSG_FLAGS	Message flags
	1...		XCG_MSG_ UPPERCASE	Uppercase msgs required
	.111 1111		*	Reserved
(63)	BITSTRING	1	*	Reserved
Parameters for Dump facilities .				
(64)	FULLWORD	4	XCG_DUMP_NUM	Dump number
(68)	ADDRESS	4	XCG_DUMP_ TITLE_PTR	Pointer to dump summary title
(6C)	FULLWORD	4	XCG_DUMP_ TITLE_LEN	Length of dump summary title
(70)	CHARACTER	8	XCG_DUMPCODE	Dumpcode
(78)	CHARACTER	9	XCG_DUMP_STR	Character form of dump id
(81)	BITSTRING	1	XCG_DUMP_FLAGS	Global dump flags
	1...		XCG_SDUMP_ IN_PROGRESS	SDUMP taken by DFHXCDMP
	.111 1111		*	Reserved
(82)	HALFWORD	2	XCG_RETRY_TIME	SDUMP Retry time
(84)	ADDRESS	4	XCG_DUMP_ ERROR_DATA	Ptr to PSW and regs for EXDUF
Pointers to TCB, XCUSER etc .				
(88)	ADDRESS	4	XCG_TCB	Pointer to our TCB
(8C)	ADDRESS	4	XCG_XCUSER_PTR	Pointer to first XCUSER block
(90)	ADDRESS	4	XCG_CURRENT_XCU	Ptr to currently inuse XCUSER
(94)	ADDRESS	4	XCG_CURRENT_XCP	Ptr to currently inuse XCPIPE
(98)	HALFWORD	2	XCG_SVC_INS	SVC number
(9A)	HALFWORD	2	*	Reserved
Timeout value from user options module				
(9C)	FULLWORD	4	XCG_TIMEOUT	Server timeout value
(A0)	CHARACTER	4	XCG_IRP_LEVEL	Returned DFHIRP level
(A4)	BITSTRING	1	XCG_IRP_ CHK_FLAGS	Returned DFHIRP level
	1...		XCG_LEVEL_ CHECKED	IRP level checked already@L1A
	.1..		XCG_LEVEL_OK	IRP level is OK
	..11 1111		*	Reserved
(A5)	BITSTRING	1	XCG_SECURITY_ FLAGS	Security options
	1...		XCG_SURROGATE_ CHK	Surrogate-user check
(A6)	BITSTRING	2	*	Reserved
Message buffer used for WTO of EXCI messages				
(A8)	CHARACTER	132	XCG_INT_MSG	Internal message area
(A8)	HALFWORD	2	XCG_INT_MSG_LEN	LL
(AA)	HALFWORD	2	XCG_INT_MSG_0	BB
(AC)	CHARACTER	124	XCG_INT_ MSG_TEXT	Maximum size msg output
(128)	FULLWORD	4	XCG_WTO_PARMS	Space for extra WTO parms
Jobname.stepname.procname string kept in XCGLOBAL, used on first DPL (as part of bind data) to inform the target CICS about who we are.				
(12C)	HALFWORD	2	XCG_JOBNAME_LEN	Length of jobname field
(12E)	CHARACTER	35	XCG_JOBNAME	Jobname field

XCUSER Block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	444	XCUSER	
(0)	CHARACTER	16	XCU_PREFIX	Standard Prefix
(0)	HALFWORD	2	XCU_LENGTH	
(2)	CHARACTER	14	XCU_EYE	>XC_USER
(10)	CHARACTER	8	XCU_APPL_NAME	Applications MYNAME
(18)	ADDRESS	4	XCU_XCG_PTR	Pointer back to XCGLOBAL
(1C)	ADDRESS	4	XCU_NEXT_XCU	Next XCUSER on chain
(20)	ADDRESS	4	XCU_PIPE_PTR	First pipe on XCUSER chain
(24)	ADDRESS	4	XCU_WS_ADDR	Pointer to PRH's working stg
(28)	CHARACTER	404	XCU_FMH07_MSG	Msg buffer returned on API
(28)	HALFWORD	2	XCU_MSG_LEN	
(2A)	HALFWORD	2	XCU_MSG_0	
(2C)	CHARACTER	400	XCU_MSG_TEXT	

XCPIPE Block

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	442	XCPIPE	
(0)	CHARACTER	16	XCP_PREFIX	Standard Prefix
(0)	HALFWORD	2	XCP_LENGTH	
(2)	CHARACTER	14	XCP_EYE	>DFHXCPIPE
(10)	ADDRESS	4	XCP_NEXT_XCP	Next pipe on the chain
(14)	CHARACTER	8	XCP_CICS_NAME	Target CICS applid
(1C)	CHARACTER	8	XCP_LOGON_NAME	Target CICS connection
(24)	ADDRESS	4	XCP_XCUSER_PTR	Pointer to owning USER block
(28)	CHARACTER	2	XCP_PIPE_STATUS	Current status of pipe
(28)	CHARACTER	1	XCP_OPEN_STATUS	Pipe is open or closed
	1...		OPEN	Pipe Open
	.1..		MUST_CLOSE	Pipe Open but must close
	..11 1111		*	Reserved
(29)	CHARACTER	1	XCP_CONV_STATE	Conversation state
	1...		FIRST_CONVERS	First convers. since open
	.111 1111		*	Reserved
(2A)	CHARACTER	2	XCP_FLAGS	
(2A)	BITSTRING	1	XCP_ALLOC_OPTS	Copy of callers alloc opts
(2B)	BITSTRING	1	*	Reserved
(2C)	ADDRESS	4	XCP_IRP_IOAREA	Addr of I/O area for IRP
(30)	UNSIGNED	4	XCP_IRP_IO_LEN	Length of I/O area
(34)	UNSIGNED	4	XCP_IRP_DLENGTH	Actual length of data sent
(38)	ADDRESS	4	XCP_XFRASTG1	Addr of Xformers I/O area
(3C)	ADDRESS	4	XCP_IRCLS	Main alist for DFHIR
(40)	CHARACTER	40	XCP_IRCSB	Sublist for DFHIR
(68)	CHARACTER	96	XCP_UU_FMH	FMH for USERID,RRS,UOWID
(C8)	CHARACTER	128	XCP_BIND	Bind data area
(148)	CHARACTER	8	LOGON_PARMS	DFHIRP LOGON parameters
(148)	ADDRESS	4	XCP_LUSERID	Logon userid
(14C)	ADDRESS	4	XCP_LSLCB	Addr of IRP's SLCB
(150)	CHARACTER	8	CONNECT_PARMS	
(150)	BITSTRING	4	XCP_THRDID	Connect thread id
(154)	ADDRESS	4	XCP_SCCB	Addr of session's SCCB
(158)	CHARACTER	32	SWITCH_PARMS	
(158)	ADDRESS	4	XCP_DATA_1	1st data address (RH)
(15C)	UNSIGNED	4	XCP_LEN_1	1st data length
(160)	ADDRESS	4	XCP_DATA_2	2nd data address (RH)
(164)	UNSIGNED	4	XCP_LEN_2	2nd data length
(168)	ADDRESS	4	XCP_DATA_3	3rd data address (RH)
(16C)	UNSIGNED	4	XCP_LEN_3	3rd data length
(170)	ADDRESS	4	XCP_DATA_4	4th data address (RH)
(174)	UNSIGNED	4	XCP_LEN_4	4th data length
(178)	CHARACTER	32	DPL_EXEC_PLIST	
(178)	ADDRESS	4	XCP_ARG_0	A(Arg0)
(17C)	ADDRESS	4	XCP_ARG_1	A(Arg1)
(180)	ADDRESS	4	XCP_ARG_2	A(Arg2)
(184)	ADDRESS	4	XCP_ARG_3	A(Arg3)
(188)	ADDRESS	4	XCP_ARG_4	A(Arg4)
(18C)	ADDRESS	4	XCP_ARG_5	A(Arg5)
(190)	ADDRESS	4	XCP_ARG_6	A(Arg6)
(194)	ADDRESS	4	XCP_ARG_7	A(Arg7)
(198)	CHARACTER	28	XCP_EID	Arg 0
(1B4)	CHARACTER	3	XCP_RH_INPUT	
(1B4)	BITSTRING	1	XCP_RH_I1	Input RH - 1st byte
(1B5)	BITSTRING	1	XCP_RH_I2	Input RH - 2nd byte
(1B6)	BITSTRING	1	XCP_RH_I3	Input RH - 3nd byte
(1B7)	CHARACTER	3	XCP_RH_OUTPUT	
(1B7)	BITSTRING	1	XCP_RH_O1	Output RH - 1st byte
(1B8)	BITSTRING	1	XCP_RH_O2	Output RH - 2nd byte
(1B9)	BITSTRING	1	XCP_RH_O3	Output RH - 3nd byte

XCTRI_PLIST - Plist for Trace Initialisation, Termination and Recovery,

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	12	XCTRI_PLIST	
(0)	UNSIGNED	1	XCTRI_FUNCTION	Function code
(1)	UNSIGNED	1	XCTRI_RESPONSE	Response code
(2)	UNSIGNED	1	* (2)	Reserved
(4)	ADDRESS	4	XCTRI_WS	A(WS for use by DFHXCTRI)
(8)	ADDRESS	4	XCTRI_XCG_PTR	A(XCGLOBAL block)

Constants

Len	Type	Value	Name	Description
14	CHARACTER	>XC_GLOBAL	XCGLOBAL_ EYECATCHER	
XCUSER Constants				
14	CHARACTER	>XC_USER	XCUSER_EYECATCHER	
XCPIPE Constants				
14	CHARACTER	>XC_PIPE	XCPIPE_EYECATCHER	
Constants for use with XCTRI_ FUNCTION				
1	HEX	01	XCTRI_INITIALISE	
1	HEX	02	XCTRI_TERMINATE	
1	HEX	03	XCTRI_RECOVERY	
Constants for use with XCTRI_ RESPONSE				
1	HEX	01	XCTRI_OK	
1	HEX	02	XCTRI_DISASTER	
External CICS Interface Abend Codes				
2	DECIMAL	401	XCSTB_CALLED_ IN_AMODE24	
2	DECIMAL	402	XCPRH_ESTAE_ SETUP_FAILURE	
2	DECIMAL	403	XCPRH_XCGLOBAL_ GM_ERROR	
2	DECIMAL	404	XCPRH_CANNOT_ CALL_XCDMP	
2	DECIMAL	405	XCPRH_SSI_ VERIFY_FAIL	
2	DECIMAL	406	XCPRH_SVC_ CALL_FAIL	
2	DECIMAL	407	XCPRH_INCORRECT_ SVC_LEVEL	
2	DECIMAL	408	XCPRH_WS_ GM_FAILURE	
2	DECIMAL	409	XCPRH_VERIFY_ GM_ERROR	
2	DECIMAL	410	XCPRH_XCUSER_ GM_FAILURE	
2	DECIMAL	411	XCDMP_NO_SVCNUM	
2	DECIMAL	412	XCEIP_UNSUPPORTED_ COMMAND	
2	DECIMAL	413	XCEIP_NO_ RETCODE_AREA	
2	DECIMAL	414	XCEIP_ESTAE_SETUP	
2	DECIMAL	415	XCEIP_CANNOT_ CALL_XCDMP	

XMANC Transaction manager domain anchor block

Transaction Manager Anchor Block
This control block contains the global storage for the
Transaction Manager domain.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	280	XMANCHOR	
(0)	CHARACTER	16	*	prefix
(0)	HALFWORD	2	XMA_LENGTH	inclusive length of anchor
(2)	CHARACTER	14	XMA_EYECATCHER	>DFHXMAnchor
(10)	CHARACTER	8	XMA_GENERAL_ SUBPOOL	XM general subpool token
(18)	ADDRESS	4	XMA_LOCK_TOKEN	XM domain lock token
(1C)	FULLWORD	4	XMA_XM_STATE	XM domain state
(20)	BITSTRING	1	XMA_GLOBAL_ USER_EXITS_STATUS	
	1...		XMA_XRSINDI_ ACTIVE	XRSINDI exit active
	.1..		XMA_XXMATT_ ACTIVE	XXMATT exit active
	..11 1111		*	Reserved
(21)	BITSTRING	1	XMA_FLAGS	Flags
	1...		XMA_FORCE_ PURGE_ISSUED	Force purge has been issued
	.1..		XMA_TXN_ WAITING_FOREVER	Some transaction is in an infinite wait due to a severe transaction initialisation or termination error
	..11 1111		*	Reserved
(22)	CHARACTER	2	*	Reserved
(24)	ADDRESS	4	XMA_CATALOG_ LOCK_TOKEN	XM domain catalog lock token

-

Transaction definition global state

Offset Hex	Type	Len	Name (Dim)	Description
(28)	CHARACTER	72	XMA_TRANDEF_ GLOBAL_STATE	
(28)	CHARACTER	24	XMA_TRANDEF_ SUBPOOL_TOKENS	
(28)	CHARACTER	8	XMA_TRANDEF_ INSTANCE_SUBPOOL	Subpool tok. for instances
(30)	CHARACTER	8	XMA_TRANDEF_ STATIC_SUBPOOL	Subpool token for static
(38)	CHARACTER	8	XMA_TRANDEF_ TPNAME_SUBPOOL	Subpool token for tpnames
(40)	CHARACTER	4	XMA_LOCAL_ SYSTEM	Sysid of local system
(44)	ADDRESS	4	XMA_STATIC_ BLOCK_HEAD	Head of static block chain
(48)	ADDRESS	4	XMA_STATIC_ BLOCK_TAIL	Tail of static block chain
(4C)	BITSTRING	1	XMA_TRANDEF_ CONTROL_FLAGS	Various control flags
	1...		XMA_TXD_ RECOVERY_COMPLETE	trandef recovery processing complete
	.111 1111		*	Reserved
(4D)	CHARACTER	3	*	Reserved
(50)	CHARACTER	12	XMA_TRANDEF_ DIRECTORY_TOKENS	
(50)	CHARACTER	4	XMA_TXD_ DIRECTORY_TOKEN	Trandef directory
(54)	CHARACTER	4	XMA_RTXD_ DIRECTORY_TOKEN	Remote trandef directory
(58)	CHARACTER	4	XMA_TPNM_ DIRECTORY_TOKEN	TPName trandef directory
(5C)	ADDRESS	4	XMA_TRANDEF_ LOCK_TOKEN	

Offset Hex	Type	Len	Name (Dim)	Description
				Trandef state lock token
(60)	UNSIGNED	4	XMA_TRANDEF_ INSTANCE_COUNT	Number of instances created
(64)	CHARACTER	8	XMA_DTRTRAN_ TOKEN	trandef token
(64)	ADDRESS	4	XMA_DTRTRAN_ TOKEN_P	trandef instance address
(68)	UNSIGNED	4	XMA_DTRTRAN_ TOKEN_N	validation number
(6C)	CHARACTER	4	XMA_DTRTRAN_ TRAN_ID	DTRTRAN tranid

--

(70)	CHARACTER	88	XMA_TRANSACTION_ GLOBAL_STATE	
(70)	FULLWORD	4	XMA_DETACH_ COUNT	number of detaches
(74)	ADDRESS	4	XMA_FIRST_ TRANSACTION	first transaction in chain
(78)	ADDRESS	4	XMA_LAST_ TRANSACTION	last transaction in chain
(7C)	ADDRESS	4	XMA_FIRST_ TXN_BROWSE	first txn browse in chain
(80)	CHARACTER	8	XMA_TRANSACTION_ SUBPOOL	transaction subpool token
(88)	ADDRESS	4	XMA_PROFORMA_ TXN	pro-forma transaction
(8C)	ADDRESS	4	XMA_FIRST_ BAD_TXN_ ENVIRONMENT	first bad txn environment (for dump formatting only)
(90)	CHARACTER	8	XMA_TRANNUM_ RANGE	trannum range
(90)	CHARACTER	4	XMA_LOW_ TRANNUM	next free trannum
(94)	CHARACTER	4	XMA_HIGH_ TRANNUM	free trannums end of range
(98)	FULLWORD	4	XMA_ATTACH_ COUNT	number of attaches
(9C)	CHARACTER	8	XMA_CSXM_ TRANDEF_TOKEN	CSXM trandef token
(A4)	CHARACTER	4	*	Reserved
(A8)	CHARACTER		*	Round to doubleword

-

TClass global state

(A8)	CHARACTER	8	XMA_TCLASS_ SUBPOOL	TClass subpool token
(B0)	CHARACTER	4	XMA_TCLASS_ DIRECTORY_TOKEN	TClass directroy token
(B4)	UNSIGNED	4	XMA_TCLASS_ INSTANCE_COUNT	Number of tclasses created
(B8)	BITSTRING	1	XMA_TCLASS_ CONTROL_FLAGS	Various control flags
	1...		XMA_TCLASS_ RECOVERY_COMPLETE	Tclass recovery processing complete
(B9)	CHARACTER	3	*	Reserved
(BC)	ADDRESS	4	XMA_TCLASS_ CHAIN_HEAD	Head of tclass master chain
(C0)	ADDRESS	4	XMA_TCLASS_ CHAIN_TAIL	Tail of tclass master chain
(C4)	CHARACTER	4	*	Reserved
(C8)	CHARACTER		*	Round to doubleword

--
-

Note that the catalogued state is placed here since MXT is the only thing that is catalogued at the moment.

MXT global state

(C8)	CHARACTER	4	XMA_CATALOGUED_ STATE	State restored from catalog
(C8)	UNSIGNED	4	XMA_MXT_LIMIT	Maximum number of user tasks
(CC)	CHARACTER	8	XMA_MXT_ TCLASS_TOKEN	

Offset Hex	Type	Len	Name (Dim)	Description
				MXT tclass token
(CC)	ADDRESS	4	XMA_MXT_ TCLASS_PTR	Address of MXT tclass
(D4)	BITSTRING	1	XMA_MXT_FLAGS	
	1...		XMA_MXT_ LIMIT_SET	MXT limit has been set
	.1..		XMA_MXT_QUEUING	System is at MXT
(D5)	CHARACTER	3	*	Reserved
(D8)	ADDRESS	4	XMA_SCHEDULER_ ERROR_HEAD	Head of queue of txns which failed in the scheduler
(DC)	ADDRESS	4	XMA_SYSTEM_ ATTACH_RETRY_HEAD	Head of queue of system txns to be re-DS attached
(E0)	FULLWORD	4	XMA_CUSHION_ SIZE_BELOW	size of 24 bit cushion
(E4)	FULLWORD	4	XMA_CUSHION_ SIZE_ABOVE	size of 31 bit cushion
(E8)	CHARACTER	8	XMA_TOTAL_TASKS	total number of tasks attached at the time of the last statistics reset
--				
(F0)	ADDRESS	4	XMA_STATS_ BUFFER_PTR	XM stats buffer address
(F4)	CHARACTER	4	*	Reserved
(F8)	CHARACTER	8	XMA_LAST_ RESET_TIME	time XM stats were last reset
(100)	CHARACTER	8	XMA_GENERAL_ SUBPOOL_24	XM general subpool token for 24 bit storage areas
(108)	CHARACTER	8	*	Spare
(110)	CHARACTER	8	XMA_RUNTRAN_ SUBPOOL	transaction subpool token for context blocks
(118)	CHARACTER		*	round to doubleword

Constants

Len	Type	Value	Name	Description
4	DECIMAL	1	PRE_INITIALISING	
4	DECIMAL	2	PRE_INITIALISED	
4	DECIMAL	3	INITIALISING	
4	DECIMAL	4	INITIALISED	
4	DECIMAL	5	QUIESCING	
4	DECIMAL	6	QUIESCED	
4	DECIMAL	7	TERMINATING	
4	DECIMAL	8	TERMINATED	

XMCAT Transaction manager catalog records

-

XM domain state catalog record

Currently the only piece of state that is saved over a CICS restart is the MXT limit.

The DTRTRAN isn't saved because no EXEC CICS SET DTRTRAN service is currently available. It is always read from the SIT so there is no need to save it over a warm start.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	4	XM_STATE_CATALOG_RECORD	
(0)	UNSIGNED	4	CAT_MXT_LIMIT	

--
-

Transaction definition catalog record.

The transaction definition externals are catalogued together with each of the aliases that the definition has. The alias existence bits indicate whether the alias names stored later in the record are actually active.

Note that the 64 character TPName is not written to the catalog in the case when the definition does not have an active TPName alias.

Both the externals and the alias information are copied directly from the transaction definition to this catalog record. The alias information is defined as a LIKE as it needs to be interpreted when the definition is recovered from the catalog. The externals are copied directly into the recovered definition and don't need to be interpreted.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	184	TRANDEF_CATALOG_RECORD	
(0)	CHARACTER	104	CAT_EXTERNALS	
(68)	CHARACTER	16	CAT_ALIASES	
(68)	BITSTRING	1	TXDSTAT_ALIAS_EXISTENCE_ BITS	
	1...		TXDSTAT_ ALIAS_X	
	.1..		TXDSTAT_ TASKREQ_X	
	..1.		TXDSTAT_ XTRANID_X	
	...1		TXDSTAT_ TPNAME_X	
	 1111		*	
(69)	CHARACTER	3	*	
(6C)	CHARACTER	4	TXDSTAT_ALIAS	
(70)	CHARACTER	4	TXDSTAT_TASKREQ	
(74)	CHARACTER	4	TXDSTAT_XTRANID	
(78)	CHARACTER	64	CAT_TPNAME	Only if active TPName

--
-

TClass catalog record.

The tclass record simply consists of the 'max-active' and 'purge-threshold' settings.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	TCLASS_CATALOG_RECORD	
(0)	UNSIGNED	4	CAT_MAX_ACTIVE	
(4)	UNSIGNED	4	CAT_PURGE_ THRESHOLD	

XMCLC Transaction manager transaction class

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	128	XM_TCLASS	
(0)	CHARACTER	16	TCL_PREFIX	
(0)	HALFWORD	2	TCL_LENGTH	Inclusive length
(2)	CHARACTER	1	TCL_ARROW	Arrow
(3)	CHARACTER	3	TCL_DFH	DFH
(6)	CHARACTER	2	TCL_DOMID	Domain-id
(8)	CHARACTER	8	TCL_BLOCK_NAME	"TCLASS " as eyecatcher
(10)	CHARACTER	8	TCL_TCLASS_NAME	Tclass name
(18)	ADDRESS	4	TCL_NEXT_TCLASS	Next tclass in master chain
(1C)	FULLWORD	4	TCL_USAGE_COUNT	No. of trandef instances referencing this tclass
(20)	FULLWORD	4	TCL_LOCK_COUNT	Number of lock requests preventing delete of tclass
(24)	CHARACTER	4	*	Reserved
(28)	CHARACTER	8	TCL_TCLASS_TOKEN	Token for this tclass
(28)	ADDRESS	4	TCL_TCLASS_ ADDRESS	Address of this tclass
(2C)	UNSIGNED	4	TCL_INSTANCE_ NUMBER	Instance validation number
(30)	CHARACTER	8	TCL_LOCK_TOKEN	Tclass resource lock token
(38)	CHARACTER	12	TCL_DEFINITION_ STATE	State of tclass definition
(38)	UNSIGNED	4	TCL_DEFINED_MAX_ACTIVE	Max. number of transactions that can be active
(3C)	UNSIGNED	4	TCL_DEFINED_PURGE_THRESHOLD	Size of queue at which transactions will be purged
(40)	BITSTRING	1	TCL_DEFINITION_ FLAGS	Various flags
	1...		TCL_DUMMY_ ENTRY	Transient dummy/placeholder tclass definition
	.1..		TCL_DUMMY_WARNING_ MSG_ISSUED	An attach-time warning msg has been issued
	..11 1111		*	Reserved
(41)	CHARACTER	3	*	Reserved
(44)	CHARACTER	60	TCL_OPERATIONAL_STATE	State of operational tclass
(44)	UNSIGNED	4	TCL_MAX_QUEUED	Maximum size of queue (one less than purge threshold except zero maps to high)
(48)	UNSIGNED	4	TCL_CURRENT_ ACTIVE	Num of txns that are active
(4C)	UNSIGNED	4	TCL_CURRENT_ QUEUED	Num of txns that are queued
(50)	ADDRESS	4	TCL_TRANSACTION_QUEUE_HEAD	Head of list of queuing txns
(54)	CHARACTER	4	*	Reserved
(58)	CHARACTER	40	TCL_STATISTICS	Statistics for this tclass
(58)	FULLWORD	4	TCL_TOTAL_ ATTACHES	Attach requests for tclass
(5C)	FULLWORD	4	TCL_PURGED_IMMEDIATELY	Purges due to purge threshold being reached
(60)	FULLWORD	4	TCL_TOTAL_ QUEUED	Txns that had to queue
(64)	FULLWORD	4	TCL_PURGED_WHILE_QUEUING	Txns purged while queuing
(68)	FULLWORD	4	TCL_PEAK_ ACTIVE	Highest number of active txns
(6C)	FULLWORD	4	TCL_PEAK_ QUEUED	Highest number of queued txns
(70)	FULLWORD	4	TCL_TIMES_AT_MAX_ACTIVE	No. of times at maxactive
(74)	FULLWORD	4	TCL_TIMES_ AT_PURGE_THRESHOLD	No. of times at purge threshold limit
(78)	CHARACTER	8	TCL_TOTAL_QUEUING_TIME	Time spent waiting by those that WERE queued
(80)	CHARACTER		*	Round to dword

XMRLC Transaction manager resource lock element

-

DFHXMRLC - Resource Lock Control Blocks

Callers of the resource locking servies must include both the resource lock element and the resource lock token control blocks.

-

Resource Lock Token

Each resource to be locked must have a double word "lock token" associated with it. The lock token must be initialised to nulls and consists of the head of the RLE chain plus an indication of the owner of the lock. If the definition is not locked then the 'owner' field will be blank.

The token must be defined on a word boundary.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	8	RESOURCE_ LOCK_TOKEN	
(0)	ADDRESS	4	RESOURCE_ LOCK_WAITERS	Waiting lock elements
(4)	BITSTRING	4	RESOURCE_ LOCK_OWNER	Identitiy of lock owner

--
-

Resource Lock Element

The Resource Lock Element describes a single waiter in a queue of tasks waiting to obtain exclusive access to a particular resource. The head of the queue is addressed from the resource lock token associated with that resource.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	17	RLE	
(0)	CHARACTER	4	RLE_EYECATCHER	>RLE as eyecatcher
(4)	ADDRESS	4	RLE_RESOURCE	Addr of resource waiting on
(8)	ADDRESS	4	RLE_NEXT	Next waiter in chain
(C)	BITSTRING	4	RLE_SUSPEND_ TOKEN	DS suspend/resume token
(10)	BITSTRING	1	RLE_FLAGS	Various flags
	1...		RLE_RESUMER	Responsibility for resume
	.111 1111		*	Reserved

XMXBC Transaction manager tran. browse element

Transaction Browse
This control block defines the transaction browse element used
to browse transactions and transaction tokens.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	32	XM_XB	
(0)	CHARACTER	16	*	prefix
(0)	HALFWORD	2	XM_XB_LENGTH	inclusive length
(2)	CHARACTER	14	XM_XB_EYECATCHER	>DFHXMTxnBrwEl
(10)	ADDRESS	4	XM_XB_NEXT_XB	next txn browse element
(14)	ADDRESS	4	XM_XB_PREV_TXN	previous transaction browsed
(18)	BITSTRING	1	XM_XB_FLAGS	flags:
	1...		XM_XB_TOKEN_ BROWSE	token browse: 0 - transaction browse, 1 - transaction token browse
(19)	UNSIGNED	1	XM_XB_TOKEN_ OWNER	owner for token browse
(1A)	CHARACTER	2	*	reserved
(1C)	ADDRESS	4	XM_XB_BROWSING_ TXN	txn which started the browse (or 0 if no such txn)
(20)	CHARACTER		*	round to doubleword

XMXDC Transaction manager transaction definition

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	200	TXD_INSTANCE	
(0)	CHARACTER	16	TXDINST_PREFIX	
(0)	HALFWORD	2	TXDINST_LENGTH	Inclusive length
(2)	CHARACTER	1	TXDINST_ARROW	Arrow
(3)	CHARACTER	3	TXDINST_DFH	DFH
(6)	CHARACTER	2	TXDINST_DOMID	Domain-id
(8)	CHARACTER	8	TXDINST_ BLOCK_NAME	"TXDINST " as eyecatcher
(10)	CHARACTER	4	TXDINST_ TRANSACTION_ID	Transid here for eyecatcher
(14)	ADDRESS	4	TXDINST_ STATIC_BLOCK_ADDR	Address of static block
(18)	ADDRESS	4	TXDINST_ BACK_CHAIN	Previous instance of this installed trandef
(1C)	CHARACTER	8	TXDINST_ TRANDEF_TOKEN	Token for this instance
(1C)	ADDRESS	4	TXDINST_ INSTANCE_ADDR	Address of this instance
(20)	UNSIGNED	4	TXDINST_ INSTANCE_NUMBER	Instance validation number
(24)	FULLWORD	4	TXDINST_ USE_COUNT	No. of txns using instance
(28)	BITSTRING	1	TXDINST_ MISCELLANEOUS_ FLAGS	Various internal flags
	1...		TXDINST_ ADD_CREATED	Instance created by Add
	.1..		TXDINST_ SET_CREATED	Instance created by Set
	..11		*	Reserved
	.... 1...		TXDINST_ SYSTEM_ATTACH	Attach as a system task
	.... .1..		TXDINST_ SHUTDOWN_OVERRIDE	Allow attaches for txn disabled at shutdown
	.... ..1.		TXDINST_DTRTRAN	Instance created as the DTRTRAN
	1		*	Reserved
(29)	UNSIGNED	1	TXDINST_REMOTE	Remote or possibly remote
(2A)	CHARACTER	2	*	Reserved
(2C)	CHARACTER	8	TXDINST_ TCLASS_TOKEN	Tclass token
(34)	CHARACTER	32	TXDINST_ TRANDEF_ RELATED_TOKENS	Owned by other areas of CICS
(34)	CHARACTER	8	TXDINST_ AP_TOKEN	AP domain's token
(3C)	CHARACTER	8	*	Reserved

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(44)	CHARACTER	8	TXDINST_ PG_TOKEN	Program Manager's token
(4C)	CHARACTER	8	*	Reserved
(54)	CHARACTER	12	*	Reserved
(60)	CHARACTER	104	TXDINST_ EXTERNALS	Users view of trandef
(60)	CHARACTER	8	TXDINST_INITIAL_PROGRAM	Initial program to invoke
(68)	CHARACTER	8	TXDINST_ PROFILE_NAME	Terminal profile to use
(70)	UNSIGNED	4	TXDINST_TWASIZE	Transaction Work Area size
(74)	UNSIGNED	1	TXDINST_ TASKDATAKEY	Taskdatakey: cics/user
(75)	UNSIGNED	1	TXDINST_ TASKDATALOC	Taskdataloc: below/any
(76)	UNSIGNED	1	TXDINST_TRAN_PRIORITY	Priority of trandef
(77)	UNSIGNED	1	TXDINST_ PARTITIONSET	Partnset: none/named/keep/own
(78)	CHARACTER	8	TXDINST_PARTITIONSET_NAME	Name of partitionset if NAMED
(80)	UNSIGNED	1	TXDINST_STATUS	Status: enabled/disabled
(81)	UNSIGNED	1	TXDINST_SYSTEM_RUNAWAY	System runaway: yes/no
(82)	UNSIGNED	1	TXDINST_ INDOUBT_WAIT	Indoubt wait: yes/no
(83)	UNSIGNED	1	TXDINST_INDOUBT_ACTION	Indoubt: backout/commit
(84)	UNSIGNED	4	TXDINST_INDOUBT_WAIT_TIME	Indoubt wait interval (mins)
(88)	UNSIGNED	4	TXDINST_RUNAWAY_LIMIT	Runaway limit if not system
(8C)	UNSIGNED	1	TXDINST_STORAGE_CLEAR	Storage clear: yes/no
(8D)	CHARACTER	1	TXDINST_ CONFDATA	Confdata: yes/no
(8E)	UNSIGNED	1	TXDINST_RESOURCE_SECURITY	Resource security: yes/no
(8F)	UNSIGNED	1	TXDINST_COMMAND_SECURITY	Command security: yes/no
(90)	UNSIGNED	4	TXDINST_ DTIMEOUT	Deadlock timeout interval
(94)	CHARACTER	8	TXDINST_ REMOTE_NAME	Txn name on remote system
(9C)	CHARACTER	4	TXDINST_REMOTE_SYSTEM	Name of remote system
(A0)	CHARACTER	8	TXDINST_TRPROF	Transaction routing profile
(A8)	UNSIGNED	1	TXDINST_DYNAMIC	Dynamic routing: yes/no
(A9)	UNSIGNED	1	TXDINST_LOCAL_QUEUING	Queue routed txns: yes/no
(AA)	UNSIGNED	1	TXDINST_STORAGE_FREEZE	Freemain storage: yes/no
(AB)	UNSIGNED	1	TXDINST_TCLASS	Txn has a TClass: yes/no
(AC)	CHARACTER	8	TXDINST_ TCLASS_NAME	TClass name if applicable
(B4)	UNSIGNED	1	TXDINST_RESTART	Transaction restart: yes/no
(B5)	UNSIGNED	1	TXDINST_SYSTEM_PURGEABLE	System purgeable: yes/no
(B6)	UNSIGNED	1	TXDINST_TERMERR_PURGEABLE	Term error purgeable: yes/no
(B7)	UNSIGNED	1	TXDINST_TRANSACTION_DUMP	Transaction dump: yes/no
(B8)	UNSIGNED	1	TXDINST_TRANSACTION_TRACE	Txn trace: stnd/specl/suprsd
(B9)	UNSIGNED	1	TXDINST_SHUTDOWN_STATUS	disabled/enabled at Shutdown
(BA)	UNSIGNED	1	TXDINST_ISOLATED_SUBSPACE	Isolated subspace: yes/no
(BB)	BITSTRING	1	TXDINST_EXTERNAL_FLAGS	Various recovered flags

Offset Hex	Type	Len	Name (Dim)	Description
	1...		TXDINST_ REMOTE_SYSTEM_SPECIFIED	RemoteSystem specified
	.111 1111		*	Reserved
(BC)	CHARACTER	8	TXDINST_BREXIT	Bridge transaction exit
(C4)	UNSIGNED	1	TXDINST_ROUTABLE_STATUS	Routable starts: routable/notroutable
(C5)	CHARACTER	3	*	Reserved
(C8)	CHARACTER		*	Round to dword

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	168	TXD_STATIC	
(0)	CHARACTER	16	TXDSTAT_PREFIX	
(0)	HALFWORD	2	TXDSTAT_LENGTH	Inclusive length
(2)	CHARACTER	1	TXDSTAT_ARROW	Arrow
(3)	CHARACTER	3	TXDSTAT_DFH	DFH
(6)	CHARACTER	2	TXDSTAT_DOMID	Domain-id
(8)	CHARACTER	8	TXDSTAT_ BLOCK_NAME	"TXDSTAT " as eyecatcher
(10)	CHARACTER	4	TXDSTAT_TRANSACTION_ID	Transaction id
(14)	ADDRESS	4	TXDSTAT_LATEST_INSTANCE	The last instance created for this definition
(18)	ADDRESS	4	TXDSTAT_NEXT_STATIC_BLOCK	Next static block in chain
(1C)	FULLWORD	4	TXDSTAT_ USE_COUNT	No. of references to this
(20)	BITSTRING	1	TXDSTAT_ STATUS_FLAGS	Various status flags
	1...		TXDSTAT_ACTIVE	Definition is active and not quiescing
	.1..		TXDSTAT_REMOTE_DIR_X	Defn. has entry in RTXD Dir
	..1.		TXDSTAT_SYSTEM_DEFINITION	Added by the system
	...1 1111		*	Reserved
(21)	CHARACTER	3	*	Reserved
(24)	ADDRESS	4	TXDSTAT_REMOTE_DIR_PREV	Prev defn with same remote name and system
(28)	ADDRESS	4	TXDSTAT_REMOTE_DIR_NEXT	Next defn with same remote name and system
(2C)	CHARACTER	8	TXDSTAT_ LOCK_TOKEN	Update lock token
(34)	CHARACTER	12	*	Reserved
(40)	CHARACTER	60	TXDSTAT_TRANDEF_STATS	Stats per installed transid
(40)	BITSTRING	8	TXDSTAT_CREATION_TIME	STCK when 1st created
(48)	UNSIGNED	4	TXDSTAT_ATTACH_COUNT	Number of attaches
(4C)	UNSIGNED	4	TXDSTAT_RESTART_COUNT	Number of restarts
(50)	UNSIGNED	4	TXDSTAT_STG_VIOLATIONS	Storage violations suffered
(54)	UNSIGNED	4	TXDSTAT_DYN_LOCAL_COUNT	Dynamic txn local runs
(58)	UNSIGNED	4	TXDSTAT_DYN_REMOTE_COUNT	Dynamic txn remote runs
(5C)	UNSIGNED	4	TXDSTAT_ REMOTE_START_COUNT	No. of remote starts of txn
(60)	UNSIGNED	4	TXDSTAT_ FORCED_ACTN_NOWAIT	No ability to wait
(64)	UNSIGNED	4	TXDSTAT_ FORCED_ACTN_OPERATOR	Forced by operator
(68)	UNSIGNED	4	TXDSTAT_ FORCED_ACTN_TIMEOUT	Forced after timeout

Offset Hex	Type	Len	Name (Dim)	Description
(6C)	UNSIGNED	4	TXDSTAT_ FORCED_ACTN_TRANDEF	Decision in trandef taken
(70)	UNSIGNED	4	TXDSTAT_ FORCED_ACTN_OTHER	Forced for other reason
(74)	UNSIGNED	4	TXDSTAT_ INDOUBT_WAIT_COUNT	Number of indoubt waits
(78)	UNSIGNED	4	TXDSTAT_ ACTION_MISMATCHES	Mismatch trandef decision
(7C)	ADDRESS	4	TXDSTAT_ TPNAME_ADDR	Addr of TPName if active
(80)	CHARACTER	4	*	Reserved
(84)	CHARACTER	16	TXDSTAT_ALIASES	
(84)	BITSTRING	1	TXDSTAT_ ALIAS_EXISTENCE_ BITS	Aliases that are active
	1...		TXDSTAT_ ALIAS_X	Alias is active
	.1..		TXDSTAT_ TASKREQ_X	Taskreq is active
	..1.		TXDSTAT_ XTRANID_X	XTranid is active
	...1		TXDSTAT_ TPNAME_X	TPName is active
	 1111		*	Reserved
(85)	CHARACTER	3	*	Reserved
(88)	CHARACTER	4	TXDSTAT_ALIAS	Alias transid if active
(8C)	CHARACTER	4	TXDSTAT_TASKREQ	Taskreq transid if active
(90)	CHARACTER	4	TXDSTAT_XTRANID	Xtranid transid if active
(94)	CHARACTER	20	TXDSTAT_ TCB_COUNTS	TCB count information
(94)	UNSIGNED	4	TXDSTAT_ NEXT_DECAY	triggers next decay
(98)	CHARACTER	8	TXDSTAT_ TOTAL_COUNTS	Current running totals
(98)	UNSIGNED	4	TXDSTAT_ TOT_ATTACHES	no. of tran attaches
(9C)	UNSIGNED	4	TXDSTAT_ TOT_TCB_COUNTS (1)	counts for TCB types
(A0)	CHARACTER	8	TXDSTAT_ INTERVAL_COUNTS	Current interval counts
(A0)	UNSIGNED	4	TXDSTAT_ INT_ATTACHES	no. of tran attaches
(A4)	UNSIGNED	4	TXDSTAT_ INT_TCB_COUNTS (1)	counts for TCB types
(A8)	CHARACTER		*	Round to dword

Constants

Len	Type	Value	Name	Description
1	DECIMAL	3	NUM_OPEN_TYPES	Number of types of open TCB which can inherit subspaces (ie DSIT_ INHERIT_YES).
1	DECIMAL	1	NUM_SUBSPACE_ OPEN_TYPES	Number of combinations of types of open TCB which can inherit subspaces (ie DSIT_ INHERIT_ YES). This number is 2 to the power NUM_ SUBSPACE_ OPEN_TYPES.
4	DECIMAL	2	COMBO_SUBSPACE_ OPEN_TYPES	

XMXNC Transaction manager transaction

Transaction
This control block defines the transaction storage for the
Transaction Manager domain.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	352	XM_TXN	
(0)	CHARACTER	16	*	prefix
(0)	HALFWORD	2	XM_TXN_LENGTH	inclusive length
(2)	CHARACTER	14	XM_TXN_ EYECATCHER	>DFHXMTxn
(10)	UNSIGNED	1	XM_TXN_ FACILITY_TYPE	facility type
(11)	UNSIGNED	1	XM_TXN_ START_CODE	start code
(12)	UNSIGNED	1	XM_TXN_ TASK_PRIORITY	task priority
(13)	BITSTRING	1	XM_TXN_FLAGS	flags
	1...		XM_TXN_ INFINITE_WAIT	transaction in infinite wait
	.1..		XM_TXN_ PRIORITY_SET	priority has been set
	..1.		XM_TXN_ INIT_PURGE_PROTECT	protected from purge during attach phase 2
	...1		XM_TXN_ TERM_PURGE_PROTECT	protected from purge during detach
	.... 1...		XM_TXN_CREATED_BY_ATTACH	created by attach rather than get txn environment
	.... .1..		XM_TXN_TCLASS	txn has a related tclass
	.... ..1.		XM_TXN_TCLASS_LOCKED	txn has a tclass locked
	1		XM_TXN_ INSUFF_STG_MSG_ISSUED	Attach failed msg issued
(14)	UNSIGNED	2	XM_TXN_ BROWSE_COUNT	# of txn browses in progress
(16)	UNSIGNED	1	XM_TXN_ATTACH_MESSAGE	attach failure message
(17)	BITSTRING	1	XM_TXN_FLAGS2	flags
	1...		XM_TXN_ DEFERRED_ABEND_TXN_DUMP	take a transaction dump on deferred abend
	.1..		XM_TXN_FORCE_PURGE_ ISSUED	force purge issued against this transaction
	..1.		XM_TXN_ PROHIBIT_INLINE_CALLS	Force inline sets to make full domain calls
	...1		XM_TXN_ DEFERRED_ABEND_SET	A deferred abend has been set
	.... 1...		XM_TXN_ DEFERRED_MESSAGE_SET	A deferred message has been set
	.... .1..		XM_TXN_ GROUP_ID_INHERITED	tran group id inherited
	.... ..1.		XM_TXN_UOW_ID_SUPPLIED	transaction is to be attached with an inherited external unit of work id
	1		XM_TXN_REPORT_CONDITION	APAC to be invoked after transaction abend
(18)	ADDRESS	4	XM_TXN_ FACILITY_TOKEN	principal_facility_address
(1C)	CHARACTER	8	XM_TXN_ PRIMARY_CLIENT_REQUEST_BLOCK	request block
(1C)	ADDRESS	4	XM_TXN_ PRIMARY_CLIENT_REQUEST_BLOCK_ADDR	address of primary client's block
(20)	FULLWORD	4	XM_TXN_ PRIMARY_CLIENT_REQUEST_BLOCK_LEN	length of primary client's block
(24)	ADDRESS	4	XM_TXN_ATTACH_PARMS_ADDR	attach parms address

Offset Hex	Type	Len	Name (Dim)	Description
(28)	FULLWORD	4	XM_TXN_ ATTACH_PARMS_LENGTH	attach parms length
(2C)	CHARACTER	8	XM_TXN_ REMOTE_NAME	remote name if applic
(34)	CHARACTER	4	XM_TXN_REMOTE_SYSTEM	remote system if applic
(38)	CHARACTER	8	XM_TXN_TRANSACTION_TOKEN	transaction token
(38)	ADDRESS	4	XM_TXN_TRANSACTION_ADDR	address of transaction
(3C)	CHARACTER	4	XM_TXN_TRANNUM	transaction number
(40)	ADDRESS	4	XM_TXN_NEXT_TRANSACTION	next transaction in chain
(44)	ADDRESS	4	XM_TXN_PREV_TRANSACTION	previous transaction in chain
(48)	CHARACTER	4	XM_TXN_ ORIGINAL_TRANSACTION_ID	original tran. id.
(4C)	CHARACTER	4	*	reserved
(50)	CHARACTER	32	*	task scheduling state
(50)	CHARACTER	8	XM_TXN_ ATTACH_TIME	XM attach time
(58)	CHARACTER	8	XM_TXN_TCLASS_WAIT_START	time TCLASS wait started
(58)	CHARACTER	8	XM_TXN_TCLASS_WAIT_TIME	time waited for TCLASS
(60)	CHARACTER	8	XM_TXN_MXT_WAIT_START	time max. task wait started
(60)	CHARACTER	8	XM_TXN_MXT_WAIT_TIME	time waited for max. task
(68)	UNSIGNED	1	XM_TXN_SCHEDULE_STAGE	stage which schedule is at
(69)	CHARACTER	3	*	reserved
(6C)	ADDRESS	4	XM_TXN_DS_TASK_TOKEN	Dispatcher task token
(70)	CHARACTER	4	XM_TXN_ PRIMARY_TRANSACTION_ID	primary tran. id.
(74)	CHARACTER	4	XM_TXN_ ABEND_CODE	abend code
(78)	UNSIGNED	1	XM_TXN_ABEND_IN_PROGRESS	abend in progress
(79)	UNSIGNED	1	XM_TXN_ SYSTEM_TRANSACTION	system transaction
(7A)	UNSIGNED	2	XM_TXN_RESTART_COUNT	restart count
(7C)	CHARACTER	4	XM_TXN_ RE_ATTACHED_UOW_TOKEN	UOW token passed by RM domain for re-attached txn resulting from an unshunt
(80)	CHARACTER	8	XM_TXN_TRANDEF_TOKEN	trandef token

-

The tokens in the XM_TXN are only ever referenced using the XMIQ set_Transaction_token and inquire_transaction_token interface. The following overlay field definitions are included only so that these fields are easily recognised in the data areas. The order of the tokens must reflect the order of the token owners defined in the CDURUN definition in DFHXMIQR e.g. xm_txn_ap_token refers to the token indexed by xmiq_ap.

Offset Hex	Type	Len	Name (Dim)	Description
(88)	CHARACTER	120	*	
(88)	CHARACTER	8	XM_TXN_TOKEN (15)	tokens
(88)	CHARACTER	120	*	
(88)	CHARACTER	8	XM_TXN_ AP_TOKEN	
(90)	CHARACTER	8	XM_TXN_ SM_TOKEN	
(98)	CHARACTER	8	XM_TXN_ TD_TOKEN	
(A0)	CHARACTER	8	XM_TXN_ MN_TOKEN	
(A8)	CHARACTER	8	XM_TXN_ PG_TOKEN	
(B0)	CHARACTER	8	*	

Offset Hex	Type	Len	Name (Dim)	Description
(B8)	CHARACTER	8	XM_TXN_ XM_TOKEN	
(C0)	CHARACTER	8	XM_TXN_ SO_TOKEN	
(C8)	CHARACTER	8	XM_TXN_ WB_TOKEN	
(D0)	CHARACTER	8	XM_TXN_ XS_TOKEN	
(D8)	CHARACTER	8	XM_TXN_ US_TOKEN	
(E0)	CHARACTER	8	XM_TXN_ LG_TOKEN	
(E8)	CHARACTER	8	XM_TXN_ TF_TOKEN	
(F0)	CHARACTER	8	XM_TXN_ RM_TOKEN	
(F8)	CHARACTER	8	XM_TXN_ BR_TOKEN	Bridge
--				
(100)	ADDRESS	4	XM_TXN_ SCHEDULER_RETRY_CHAIN	System DS attaches to retry
(100)	ADDRESS	4	XM_TXN_ SCHEDULER_ERROR_CHAIN	Txns with fatal errors in scheduler
(104)	CHARACTER	16	*	Tclass state
(104)	ADDRESS	4	XM_TXN_TCLASS_DELAY_ADDR	Addr of area to store queuing delay
(108)	ADDRESS	4	XM_TXN_ NEXT_TCLASS_WAITER	Next transaction waiting for tclass/MXT
(10C)	CHARACTER	8	XM_TXN_ TCLASS_TOKEN	tclass token
(114)	CHARACTER	4	XM_TXN_DEFERRED_ABEND	deferred abend code
(118)	CHARACTER	27	XM_TXN_EXTERNAL_UOW_ID	SNA architected unit of work id
(133)	UNSIGNED	1	XM_TXN_ RE_ATTACHED_TRANSACTION	Re-attached txn as a result of RM domain unshunt
(134)	UNSIGNED	1	XM_TXN_ ROLLBACK_REQUESTED	Commit to be converted to Rollback
(135)	UNSIGNED	1	XM_TXN_RESTART	transaction is to be restarted after transaction abend
(136)	CHARACTER	6	*	reserved
(13C)	BITSTRING	1	XM_TXN_ROUTABLE_STATUS	transaction routable status
(13D)	BITSTRING	1	XM_TXN_ PRIMARY_CLIENT_TYPE	identity of component that issued the ATTACH
(13E)	CHARACTER	28	XM_TXN_ TRANSACTION_GROUP_ID	transaction group id
(160)	CHARACTER		*	round to doubleword

Constants

Len	Type	Value	Name	Description
1	DECIMAL	0	XM_TXN_NULL_ATTACH_MESSAGE	
Values for xm_ txn_primary_ client_type				
1	DECIMAL	1	XM_TXN_NONE	
1	DECIMAL	2	XM_TXN_TERMINAL	
1	DECIMAL	3	XM_TXN_TRANDATA	
1	DECIMAL	4	XM_TXN_START	
1	DECIMAL	5	XM_TXN_START_ TERMINAL	
1	DECIMAL	6	XM_TXN_SCHEDULER	
1	DECIMAL	7	XM_TXN_XM_RUN_TRANSACTION	
1	DECIMAL	8	XM_TXN_BRIDGE	
1	DECIMAL	9	XM_TXN_SOCKET	
1	DECIMAL	10	XM_TXN_WEB	
1	DECIMAL	11	XM_TXN_IIOP	
1	DECIMAL	12	XM_TXN_RRS_UR	
1	DECIMAL	13	XM_TXN_LU61_ SESSION	
1	DECIMAL	14	XM_TXN_APPC_ SESSION	
1	DECIMAL	15	XM_TXN_MRO_SESSION	
Values for xm_ txn_schedule_ stage				
1	DECIMAL	1	XM_TXN_PRE_ SCHEDULE	
1	DECIMAL	2	XM_TXN_TCLASS_SCHEDULED	
1	DECIMAL	3	XM_TXN_MXT_ SCHEDULED	
1	DECIMAL	4	XM_TXN_DS_ATTACHED	
Null value for xm_ txn_deferred_ abend				

Len	Type	Value	Name	Description
4	CHARACTER		XM_TXN_NULL_ DEFERRED_ABEND	

declare xm_ txn_null_ token char(8) constant('0000000000000000'x);
The following constant must be used until all the users of
DFHXMCON are converted to PL/X

Len	Type	Value	Name	Description
4	DECIMAL	0	XM_TXN_NULL_TOKEN	
4	DECIMAL	15	XM_TXN_TOKEN_ OWNERS	

XSANC Security domain anchor block

-

Define the XS Domain declarations. This step produces the "DFHXSANC COPY" file, for general use by the domain. This copybook also contains constants required by all the modules in the domain.

Note that this copy file will be used in other routines, for example DFHXSTRI for trace interpretation.

Because this file uses the user-defined types declared in "DFHXSTYP COPY", all programs that include this file must also include "DFHXSTYP".

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	64	XSA	XS domain anchor block
(0)	CHARACTER	16	XSA_PREFIX	===> eyecatcher <===
(0)	HALFWORD	2	XSA_PREFIX_ LENGTH	length of xsa
(2)	CHARACTER	14	XSA_PREFIX_TEXT	>DFHXSANCHOR
(10)	UNSIGNED	1	XSA_XS_STATE	XS domain state initialised, quiesce or terminated
(11)	BITSTRING	1	*	reserved for flags
(12)	CHARACTER	2	XSA_CICS_SVC	The CICS type-3 SVC
(12)	UNSIGNED	1	XSA_CICS_ SVC_OPCODE	SVC operation code
(13)	UNSIGNED	1	XSA_CICS_ SVC_NUMBER	SVC number from kernel
(14)	ADDRESS	4	XSA_AUTHORIZED_ BLOCK_POINTER	The key-zero portion of the XS state
(18)	CHARACTER	4	XSA_APPC_SEED	"Random Number" seed for XSLU APPC Functions

-

Here we define the subpool tokens representing the various storage manager subpools acquired for the Security Domain.

Offset Hex	Type	Len	Name (Dim)	Description
(1C)	STRUCTURE IsA(ETOKEN)	8	XSA_SPTOKEN_ GENERAL	General use subpool, including the XS anchor
(1C)	ADDRESS	4	P	
(20)	FULLWORD	4	N	
(24)	STRUCTURE IsA(ETOKEN)	8	XSA_XSXM_POOL	Quickcell pool for XSXM
(24)	ADDRESS	4	P	
(28)	FULLWORD	4	N	

--
-

Here we define the lock tokens representing the various locks obtained from the Lock Manager and used by the Security Domaain.

Offset Hex	Type	Len	Name (Dim)	Description
(2C)	ADDRESS	4	XSA_DOMAIN_ LOCK_TOKEN	XS domain lock token
(30)	ADDRESS	4	XSA_RESCHECK_ LOCK_TOKEN	Resource check lock
(34)	ADDRESS	4	XSA_REBUILD_ LOCK_TOKEN	Security Rebuild lock
(38)	ADDRESS	4	XSA_EXTRACT_ LOCK_TOKEN	Security Extract lock

Offset Hex	Type	Len	Name (Dim)	Description
--				
(40)	CHARACTER		*	Reserved This is for double word boundary alignment. End of XS anchor block

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	XS_STATE_ INITIALISING	
1	DECIMAL	2	XS_STATE_ INITIALISED	
1	DECIMAL	3	XS_STATE_QUIESCING	
1	DECIMAL	4	XS_STATE_QUIESCED	
1	DECIMAL	5	XS_STATE_ TERMINATED	
	Component id (for use on ME domain calls)			
2	CHARACTER	XS	COMPID	used on ME domain call
	Standard message numbers and system dumpcode values			
1	DECIMAL	1	MNO_ABEND	
8	CHARACTER	XS0001	DCD_ABEND	
1	DECIMAL	2	MNO_SEVERE_ERROR	
8	CHARACTER	XS0002	DCD_SEVERE_ERROR	
1	DECIMAL	3	MNO_NO_STORAGE	
8	CHARACTER	XS0003	DCD_NO_STORAGE	
1	DECIMAL	4	MNO_LOOP	
8	CHARACTER	XS0004	DCD_LOOP	
1	DECIMAL	5	MNO_STCK_ERROR	
8	CHARACTER	XS0005	DCD_STCK_ERROR	
1	DECIMAL	6	MNO_NO_MVS_STORAGE	
8	CHARACTER	XS0006	DCD_NO_MVS_STORAGE	
	XS domain message numbers and system dumpcode values			
4	DECIMAL	1108	MNO_APPCLU_ RACLIST_FAILED	
8	CHARACTER	XS1108	DCD_APPCLU_ RACLIST_FAILED	
	Trace point identifiers			
2	HEX	0101	TID_XSDM_ENTRY	
2	HEX	0102	TID_XSDM_EXIT	
2	HEX	0103	TID_XSDM_RECOVERY	
2	HEX	0104	TID_XSDM_ INVALID_FORMAT	
2	HEX	0105	TID_XSDM_ INVALID_FUNCTION	
2	HEX	0106	TID_XSDM_ LOCK_ERROR	
2	HEX	0107	TID_XSDM_ UNLOCK_ERROR	
2	HEX	0108	TID_XSDM_ NO_STORAGE_FOR_ XSA	
2	HEX	0109	TID_XSDM_ GET_PARMS_FAILED	
2	HEX	010A	TID_XSDM_ GET_SVC_ERROR	
2	HEX	0201	TID_XSAD_ENTRY	
2	HEX	0202	TID_XSAD_EXIT	
2	HEX	0203	TID_XSAD_RECOVERY	
2	HEX	0204	TID_XSAD_ INVALID_FORMAT	
2	HEX	0205	TID_XSAD_ INVALID_FUNCTION	
2	HEX	0206	TID_XSAD_ XSSA_FAILURE	
2	HEX	0207	TID_XSAD_ XSSB_FAILURE	
2	HEX	0301	TID_XSIS_ENTRY	
2	HEX	0302	TID_XSIS_EXIT	
2	HEX	0303	TID_XSIS_RECOVERY	
2	HEX	0304	TID_XSIS_ INVALID_FORMAT	
2	HEX	0305	TID_XSIS_ INVALID_FUNCTION	
2	HEX	0306	TID_XSIS_ XSSC_FAILURE	
2	HEX	0307	TID_XSIS_ XSSI_FAILURE	
2	HEX	0308	TID_XSIS_ EXTRACT_LOCK_ERROR	
2	HEX	0309	TID_XSIS_ EXTRACT_UNLOCK_ ERROR	
2	HEX	030A	TID_XSIS_ REBUILD_LOCK_ERROR	
2	HEX	030B	TID_XSIS_ REBUILD_UNLOCK_ ERROR	
2	HEX	0401	TID_XSXM_ENTRY	
2	HEX	0402	TID_XSXM_EXIT	
2	HEX	0403	TID_XSXM_RECOVERY	
2	HEX	0404	TID_XSXM_ INVALID_FORMAT	

Len	Type	Value	Name	Description
2	HEX	0405	TID_XSXM_ INVALID_FUNCTION	
2	HEX	0406	TID_XSXM_ GETMAIN_FAILURE	
2	HEX	0501	TID_XSFL_ENTRY	
2	HEX	0502	TID_XSFL_EXIT	
2	HEX	0503	TID_XSFL_RECOVERY	
2	HEX	0504	TID_XSFL_ INVALID_FORMAT	
2	HEX	0505	TID_XSFL_ INVALID_FUNCTION	
2	HEX	0506	TID_XSFL_ INVALID_SECURITY_ TOKEN	
2	HEX	0507	TID_XSFL_ INVALID_FORMAT_ PASSED_TO_XSSA	
2	HEX	0508	TID_XSFL_ INVALID_FUNCTION_ PASSED_TO_XSSA	
2	HEX	0509	TID_XSFL_ INVALID_FLATTENED_ BUFFER	
2	HEX	050A	TID_XSFL_ DISASTROUS_ERROR_ IN_XSSA	
2	HEX	0601	TID_XSPW_ENTRY	
2	HEX	0602	TID_XSPW_EXIT	
2	HEX	0603	TID_XSPW_RECOVERY	
2	HEX	0604	TID_XSPW_ INVALID_FORMAT	
2	HEX	0605	TID_XSPW_ INVALID_FUNCTION	
2	HEX	0606	TID_XSPW_ XSSB_FAILURE	
2	HEX	0607	TID_XSPW_ XSSD_FAILURE	
2	HEX	0608	TID_XSPW_ XSSE_FAILURE	
2	HEX	0701	TID_XSRC_ENTRY	
2	HEX	0702	TID_XSRC_EXIT	
2	HEX	0703	TID_XSRC_RECOVERY	
2	HEX	0704	TID_XSRC_ INVALID_FORMAT	
2	HEX	0705	TID_XSRC_ INVALID_FUNCTION	
2	HEX	0706	TID_XSRC_ LOCK_ERROR	
2	HEX	0707	TID_XSRC_ UNLOCK_ERROR	
2	HEX	0708	TID_XSRC_ DISPATCHER_ERROR	
2	HEX	0709	TID_XSRC_ RESOURCE_CHECK_ ENTRY	
2	HEX	070A	TID_XSRC_ RESOURCE_CHECK_ EXIT	
2	HEX	070B	TID_XSRC_ RESOURCE_CHECK_ ERROR	
2	HEX	070C	TID_XSRC_ INVALID_RESOURCE_ TYPE	
2	HEX	070D	TID_XSRC_ INVALID_ACCESS	
2	HEX	070E	TID_XSRC_ XSSC_FAILURE	
2	HEX	070F	TID_XSRC_ XRF_TRACKING_ERROR	
2	HEX	0801	TID_XSLU_ENTRY	
2	HEX	0802	TID_XSLU_EXIT	
2	HEX	0803	TID_XSLU_RECOVERY	
2	HEX	0804	TID_XSLU_ INVALID_FORMAT	
2	HEX	0805	TID_XSLU_ INVALID_FUNCTION	
2	HEX	0806	TID_XSLU_ ESTAE_FAILURE	
2	HEX	0807	TID_XSLU_ EXTRACT_FAILURE	
2	HEX	0808	TID_XSLU_ XSSB_FAILURE	
2	HEX	0809	TID_XSLU_ EXTRACT_LOCK_ERROR	
2	HEX	080A	TID_XSLU_ EXTRACT_UNLOCK_ ERROR	
2	HEX	FE01	TID_XSS_ENTRY	
2	HEX	FE02	TID_XSS_EXIT	
2	HEX	FE03	TID_XSS_INSTALLATION_ DATA	
2	HEX	FE04	TID_XSS_EXCEPTION	
2	HEX	FE05	TID_XSS_SVC_ERROR	
	Subpool Names			
8	CHARACTER	XSGENRAL	SPNAME_GENERAL	
8	CHARACTER	XSXMPOOL	XSXM_SUBPOOL_NAME	

Len	Type	Value	Name	Description
	Anchor block eyecatcher			
14	CHARACTER	>DFHXSANCHOR	XSA_EYE_CATCHER	
	Security Lock names			
8	CHARACTER	XSLOCK	XS_DOMAIN_LOCKNAME	
8	CHARACTER	XSRCHECK	XS_RESCHECK_ LOCKNAME	
8	CHARACTER	XSRBUILD	XS_REBUILD_ LOCKNAME	
8	CHARACTER	XSXTRACT	XS_EXTRACT_ LOCKNAME	

XSSS Security supervisor storage

-

Security domain supervisor storage.

This is the storage area managed by the Security Domain's SVC routine, DFHXSS.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	288	DFHXSSS	Security supervisor storage
(0)	CHARACTER	16	XSSS_EYECATCHER	Standard control block prefix
(0)	HALFWORD	2	XSSS_LENGTH	Length of entire control block
(2)	CHARACTER	1	XSSS_ARROW	Highlighting arrow
(3)	CHARACTER	5	XSSS_COMPONENT	Component identification
(8)	CHARACTER	8	XSSS_BLOCKID	Block identification
(10)	UNSIGNED	1	XSSS_VERSION	Version number of block
(11)	CHARACTER	1	XSSS_FLAG1	Security Domain flag byte 1
	1...		XSSS_SECURITY_ ACTIVE	Security active....SEC=YES
	.1..		XSSS_PREFIX_ REQUIRED	Prefixing active...SECPRFX=YES
	..1.		XSSS_SURROGATE_ CHECK	Surrogate checking.XUSER=YES
	...1		XSSS_PARTNER_ CHECK	Partner LU check...XAPPC=YES
	.... 1...		XSSS_INSTLN_ REQUIRED	ESM instln data....ESMEXITS=
	.... .1..		XSSS_PSB_CHECK	PSB check..........PSBCHK=YES
	11		*	Reserved
(12)	CHARACTER	1	XSSS_FLAG2	Security Domain flag byte 2
(12)	BITSTRING	1	*	Reserved
(13)	CHARACTER	1	XSSS_FLAG3	Security Domain flag byte 3
	1...		XSSS_RESSEC	Always perform RESSEC
	.1..		XSSS_CMDSEC	Always perform CMDSEC
	..11 1111		*	Reserved
(14)	ADDRESS	4	XSSS_CWA_ADDRESS	CWA address (only if ESMEXITS=INSTLN)
(18)	CHARACTER	8	XSSS_SUBSYS	CICS subsystem identifier

-

This section contains pointers to various service routines that are required to be in protected storage for integrity reason.

Offset Hex	Type	Len	Name (Dim)	Description
(20)	CHARACTER	16	XSSS_SECURITY_ VECTOR_TABLE	Miscellaneous pointers
(20)	ADDRESS	4	XSSS_EARLY_ VERIFY_ROUTINE	Early verification stub
(24)	ADDRESS	4	*	Reserved
(28)	ADDRESS	4	*	Reserved
(2C)	ADDRESS	4	*	Reserved

--

Offset Hex	Type	Len	Name (Dim)	Description
(30)	STRUCTURE IsA(SECURITY_TOKEN)	8	XSSS_DEFAULT_ SECURITY_TOKEN	Token for default user
(30)	ADDRESS	4	P	
(34)	FULLWORD	4	N	
(38)	STRUCTURE IsA(SECURITY_TOKEN)	8	XSSS_JOBSTEP_ SECURITY_TOKEN	Token for jobstep user
(38)	ADDRESS	4	P	

Offset Hex	Type	Len	Name (Dim)	Description
(3C)	FULLWORD	4	N	

-

APPCLU Filter String

We supply the ESM with a filter so that only those profiles relevant to our CICS Region's VTAM netid and local LUname are brought into storage.

This filter is build after CICS opens the VTAM ACB, which may occur a long time after CICS has initialised.

The filter is built with a 2 byte length prefix to meet the requirements of the ESM.

This filter is only built if the SIT specified XAPPC=YES.

Offset Hex	Type	Len	Name (Dim)	Description
(40)	CHARACTER	24	XSSS_APPCLU_ FILTER	Used in RACLIST processing
(40)	HALFWORD	2	XSSS_APPCLU_FILTER_LENGTH	actual length of string
(42)	CHARACTER	22	XSSS_APPCLU_FILTER_STRING	= netid.local_luname.*

--

Offset Hex	Type	Len	Name (Dim)	Description
(58)	CHARACTER	8	XSSS_GENERIC_ APPLID	Generic applid for region
(60)	FULLWORD	4	*	Reserved for alignment
(64)	HALFWORD	2	XSSS_CLASSNAME_COUNT	Number of entries in the classname table
(66)	CHARACTER	1	*	Reserved
(67)	STRUCTURE IsA(USERID)	11	XSSS_REGION_ USERID	Userid for CICS region
(67)	UNSIGNED	1	LEN	
(68)	CHARACTER	10	VAL	
(72)	CHARACTER	5	*	Reserved
(77)	STRUCTURE IsA(GROUPID)	11	XSSS_REGION_ GROUPID	Groupid for CICS region
(77)	UNSIGNED	1	LEN	
(78)	CHARACTER	10	VAL	
(82)	CHARACTER	5	*	Reserved
(87)	STRUCTURE IsA(PREFIX)	11	XSSS_PREFIX	Resource name prefix
(87)	UNSIGNED	1	LEN	
(88)	CHARACTER	10	VAL	
(92)	HALFWORD	2	*	Reserved for alignment

-

This section contains the anchor blocks for the various management routines used to allocate and use security tokens.

Offset Hex	Type	Len	Name (Dim)	Description
(94)	CHARACTER	20	XSSS_SECURITY_TOKEN_MANAGER	Security token manager
(94)	ADDRESS	4	XSSS_DIRECTORY_ PTR	Directory manager anchor
(98)	ADDRESS	4	XSSS_STORAGE_INTERFACE_PTR	Storage interface anchor
(9C)	ADDRESS	4	XSSS_STORAGE_MANAGER_PTR	Storage manager anchor
(A0)	ADDRESS	4	XSSS_EXTENSION_MANAGER_PTR	Storage extension anchor
(A4)	UNSIGNED	4	XSSS_TOKEN_HWMK	Allocation high-water-mark

--

Offset Hex	Type	Len	Name (Dim)	Description
(A8)	CHARACTER	120	XSSS_CLASSNAME_ TABLE	Classnames
(A8)	CHARACTER	10	XSSS_APPC	XAPPC entry
(A8)	CHARACTER	8	CLASS_NAME	
(B0)	CHARACTER	1	CLASS_FLAGS	
	1...		CLASS_RESSEC	
	.1..		CLASS_CMDSEC	
	..11 11..		*	
	.... ..1.		CLASS_REBUILD	
	1		CLASS_ACTIVE	
(B1)	UNSIGNED	1	CLASS_MEMBER_LENGTH	
(B2)	CHARACTER	10	XSSS_TRANSACTION	XPCT entry

Offset Hex	Type	Len	Name (Dim)	Description
(B2)	CHARACTER	8	CLASS_NAME	
(BA)	CHARACTER	1	CLASS_FLAGS	
	1...		CLASS_RESSEC	
	.1..		CLASS_CMDSEC	
	..11 11..		*	
	.... ..1.		CLASS_REBUILD	
	1		CLASS_ACTIVE	
(BB)	UNSIGNED	1	CLASS_MEMBER_LENGTH	
(BC)	CHARACTER	10	XSSS_SPCOMMAND	XCMD entry
(BC)	CHARACTER	8	CLASS_NAME	
(C4)	CHARACTER	1	CLASS_FLAGS	
	1...		CLASS_RESSEC	
	.1..		CLASS_CMDSEC	
	..11 11..		*	
	.... ..1.		CLASS_REBUILD	
	1		CLASS_ACTIVE	
(C5)	UNSIGNED	1	CLASS_MEMBER_LENGTH	
(C6)	CHARACTER	10	XSSS_DB2ENTRY	XDB2ENT entry
(C6)	CHARACTER	8	CLASS_NAME	
(CE)	CHARACTER	1	CLASS_FLAGS	
	1...		CLASS_RESSEC	
	.1..		CLASS_CMDSEC	
	..11 11..		*	
	.... ..1.		CLASS_REBUILD	
	1		CLASS_ACTIVE	
(CF)	UNSIGNED	1	CLASS_MEMBER_LENGTH	
(D0)	CHARACTER	10	XSSS_TDQUEUE	XDCT entry
(D0)	CHARACTER	8	CLASS_NAME	
(D8)	CHARACTER	1	CLASS_FLAGS	
	1...		CLASS_RESSEC	
	.1..		CLASS_CMDSEC	
	..11 11..		*	
	.... ..1.		CLASS_REBUILD	
	1		CLASS_ACTIVE	
(D9)	UNSIGNED	1	CLASS_MEMBER_LENGTH	
(DA)	CHARACTER	10	XSSS_FILE	XFCT entry
(DA)	CHARACTER	8	CLASS_NAME	
(E2)	CHARACTER	1	CLASS_FLAGS	
	1...		CLASS_RESSEC	
	.1..		CLASS_CMDSEC	
	..11 11..		*	
	.... ..1.		CLASS_REBUILD	
	1		CLASS_ACTIVE	
(E3)	UNSIGNED	1	CLASS_MEMBER_LENGTH	
(E4)	CHARACTER	10	XSSS_JOURNAL	XJCT entry
(E4)	CHARACTER	8	CLASS_NAME	
(EC)	CHARACTER	1	CLASS_FLAGS	
	1...		CLASS_RESSEC	
	.1..		CLASS_CMDSEC	
	..11 11..		*	
	.... ..1.		CLASS_REBUILD	
	1		CLASS_ACTIVE	
(ED)	UNSIGNED	1	CLASS_MEMBER_LENGTH	
(EE)	CHARACTER	10	XSSS_PROGRAM	XPPT entry
(EE)	CHARACTER	8	CLASS_NAME	
(F6)	CHARACTER	1	CLASS_FLAGS	
	1...		CLASS_RESSEC	
	.1..		CLASS_CMDSEC	
	..11 11..		*	
	.... ..1.		CLASS_REBUILD	
	1		CLASS_ACTIVE	
(F7)	UNSIGNED	1	CLASS_MEMBER_LENGTH	
(F8)	CHARACTER	10	XSSS_PSB	XPSB entry
(F8)	CHARACTER	8	CLASS_NAME	
(100)	CHARACTER	1	CLASS_FLAGS	
	1...		CLASS_RESSEC	
	.1..		CLASS_CMDSEC	
	..11 11..		*	
	.... ..1.		CLASS_REBUILD	
	1		CLASS_ACTIVE	
(101)	UNSIGNED	1	CLASS_MEMBER_LENGTH	
(102)	CHARACTER	10	XSSS_TSQUEUE	XTST entry
(102)	CHARACTER	8	CLASS_NAME	
(10A)	CHARACTER	1	CLASS_FLAGS	
	1...		CLASS_RESSEC	
	.1..		CLASS_CMDSEC	
	..11 11..		*	
	.... ..1.		CLASS_REBUILD	

Offset Hex	Type	Len	Name (Dim)	Description
	1		CLASS_ACTIVE	
(10B)	UNSIGNED	1	CLASS_MEMBER_ LENGTH	
(10C)	CHARACTER	10	XSSS_TRANSATTACH	XTRAN entry
(10C)	CHARACTER	8	CLASS_NAME	
(114)	CHARACTER	1	CLASS_FLAGS	
	1...		CLASS_RESSEC	
	.1..		CLASS_CMDSEC	
	..11 11..		*	
	.... ..1.		CLASS_REBUILD	
	1		CLASS_ACTIVE	
(115)	UNSIGNED	1	CLASS_MEMBER_ LENGTH	
(116)	CHARACTER	10	XSSS_SURROGATE	XUSER entry
(116)	CHARACTER	8	CLASS_NAME	
(11E)	CHARACTER	1	CLASS_FLAGS	
	1...		CLASS_RESSEC	
	.1..		CLASS_CMDSEC	
	..11 11..		*	
	.... ..1.		CLASS_REBUILD	
	1		CLASS_ACTIVE	
(11F)	UNSIGNED	1	CLASS_MEMBER_ LENGTH	
(120)	CHARACTER		XSSS_CLASSNAME_ TABLE_END	End of table
(120)	CHARACTER		*	Reserved for alignment
(120)	CHARACTER		*	Reserved for alignment

-

Resource class table entry

The following is an entry in the resource class table.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	10	CLASSENTRY	Entry in resource class table
(0)	CHARACTER	8	CLASS_NAME	ESM classname for this entry
(8)	CHARACTER	1	CLASS_FLAGS	Flags
	1...		CLASS_RESSEC	This class subject to RESSEC
	.1..		CLASS_CMDSEC	This class subject to CMDSEC
	..11 11..		*	Reserved
	.... ..1.		CLASS_REBUILD	This class being rebuilt
	1		CLASS_ACTIVE	This class is RACLISTed
(9)	UNSIGNED	1	CLASS_MEMBER_ LENGTH	Maximum member length

--
-

Security Directory entry

The following is an entry in the Security Domain's directory. It is located from a Security_Token by using BPQHSH2 Building Block that is anchored in "xsss_directory_ptr." Note that, to save storage, "xsdi_applid" is only present if its existence bit ("xsdi_applid_x") is set.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	40	XSDI_SECURITY_ ENTRY	Security Entry
(0)	HALFWORD	2	XSDI_LENGTH	Length of entry
(2)	BITSTRING	1	XSDI_FLAGS	Flag byte
	1...		*	Reserved
	.1..		XSDI_APPLID_X	Applid is present
	..11 1111		*	Reserved
(3)	STRUCTURE IsA(USERID)	11	XSDI_USERID	Owning userid
(3)	UNSIGNED	1	LEN	
(4)	CHARACTER	10	VAL	
(E)	BITSTRING	1	*	Reserved for alignment

Offset Hex	Type	Len	Name (Dim)	Description
(F)	STRUCTURE IsA(ENTRY_PORT)	9	XSDI_ENTRY_PORT	Associated Port-of-Entry
(F)	UNSIGNED	1	TYPE	
(10)	CHARACTER	8	NAME	
(18)	ADDRESS	4	XSDI_ACEE_PTR	Address of ACEE
(1C)	UNSIGNED	4	*	Reserved
(20)	CHARACTER	8	XSDI_APPLID	(Optional) applid

Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	XSSS_V321	Version 3.2.1
1	DECIMAL	2	XSSS_V410	Version 4.1.0
1	DECIMAL	2	XSSS_VERSION_NUM	Current version

The following constant defines the length of the flattened security data block. This length must be the same as that defined in DFHXSSA. If it is not, DFHXSSA will not compile.

1	DECIMAL	48	XSSS_FLATTENED_ SECURITY_LENGTH	

XSXD Security domain transaction data

-

There is one such structure for every transaction.

The structure contains the three types of facility token expressed first as a three-element array, and then as individually named tokens. All the unique instances of these tokens are kept in another three element array.

Offset Hex	Type	Len	Name (Dim)	Description
(0)	STRUCTURE	56	XSXD_TRANSACTION_ DATA	
(0)	CHARACTER	24	*	
(0)	CHARACTER	8	XSXD_FACILITY_ TOKEN (3)	
(0)	ADDRESS	4	P	
(4)	FULLWORD	4	N	
(0)	CHARACTER	24	*	
(0)	CHARACTER	8	XSXD_PRINCIPAL_ TOKEN	
(0)	ADDRESS	4	P	
(4)	FULLWORD	4	N	
(8)	CHARACTER	8	XSXD_SESSION_ TOKEN	
(8)	ADDRESS	4	P	
(C)	FULLWORD	4	N	
(10)	CHARACTER	8	XSXD_EDF_TOKEN	
(10)	ADDRESS	4	P	
(14)	FULLWORD	4	N	
(18)	CHARACTER	24	XSXD_UNIQUE_ TOKEN_LIST	
(18)	CHARACTER	8	XSXD_UNIQUE_ TOKEN (3)	
(18)	ADDRESS	4	P	
(1C)	FULLWORD	4	N	

-

We also include a double-word communication area, which is intended for communication between the early-verification phase of the signon function and the normal verification phase, entered during ADD_ USER security processing. This double-word is only used by non-RACF external security managers, and is never used by CICS.

(30)	BITSTRING	8	XSXD_COMMUNICATION_ AREA	

XSXT Security domain transaction token

-

This structure defines the format of the Security Domain transaction token that is preserved by the Transaction Manager. There is one such token for each transaction.

The transaction token consists of two fullwords. The first fullword is the address of the transaction data. The second fullword contains a 16-bit stack of transaction options, that is, eight pairs of RESSEC and CMDSEC options. The topmost pair represent the current RESSEC and CMDSEC. The low-order 16 bits are reserved for a count of the stack depth, but it is not currently used.

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(0)	STRUCTURE	8	XSXT_TRAN_TOKEN	XS Transaction token
(0)	ADDRESS	4	XSXT_TRAN_ DATA_PTR	Ptr to transaction data
(4)	BITSTRING	2	XSXT_STACK	Stack of RESSEC/CMDSEC
(4)	BITSTRING	1	XSXT_STACK_1	First byte of stack
	`1... ....`		XSXT_RESSEC	Current RESSEC value
	`.1.. ....`		XSXT_CMDSEC	Current CMDSEC value
(5)	BITSTRING	1	XSXT_STACK_2	Second byte of stack
(6)	HALFWORD	2	XSXT_COUNT	Not used

ZCQ Builder services action blocks

CONTROL BLOCK NAME = DFHTBSGC
DESCRIPTIVE NAME = CICS Table Builder Services Action Blocks
FUNCTION =

DFHTBSGC describes the dsect for Builder Services Action Blocks. These blocks are arrays of elements that describe the actions taken to Install , Delete, Recover or Catalog communication resource definitions.
BS Action Blocks are hung of either the Resource definition Recovery Anchor Block (RRAB) (for those that either relate to general resources or have been moved onto the delayed_action_list prior to commitment/rollback), or from a Resource defintion Atom Block (RABN) (because they are for a named atom).
They are created by Table Builder Services when a request starts and are filled and/or added to the chain when Builder modules are driven. The log record that relates to a particular builders activity is chained from the relevant action element.
The Table Builder Services Syncpoint program DFHTBSS frees the action_blocks once they have been used at the end of the Builder Services Request (often at Syncpoint)

LIFETIME =
For the duration of the Table Builder Services Request
STORAGE CLASS =
Above 16M line. CICS key.
LOCATION =
Chained from the RRAB or one of the RABNs on the RRABs chain of named atoms.
INNER CONTROL BLOCKS = None
NOTES :
DEPENDENCIES = S/370
RESTRICTIONS = None
MODULE TYPE = Control block definition

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(0)	STRUCTURE	*	BS_ACTION	
(0)	ADDRESS	4	BS_ACTION_NEXT	Next action this ATOM
(4)	ADDRESS	4	BS_ACTION_PREV	Previous action this ATOM
(8)	CHARACTER	16	BS_ACTION_ID	Ident >DFHBS_ACTION_BK
(18)	CHARACTER	8	BS_ACTION_PLM	Name of module for builder
(20)	ADDRESS	4	BS_ACTION_REQSTG	Request-unique storage

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(24)	UNSIGNED	2	BS_ACTION_MSIZE	Max number of elements
(26)	UNSIGNED	2	BS_ACTION_CSIZE	Current number of elems
(28)	UNSIGNED	1	*	Reserved
The following field is an array of BS_ACTION_ELEMENTS				
(29)	CHARACTER	13	BS_ACTION_ARRAY (*)	

This is the layout of each action element BS_ ACTION_ELEM

Offset Hex	**Type**	**Len**	**Name (Dim)**	**Description**
(0)	STRUCTURE	13	BS_ACTION_ELEM	
(0)	ADDRESS	4	BS_ACTION_PATT	Address of pattern
(4)	ADDRESS	4	BS_ACTION_NODE	Node for this action
(8)	ADDRESS	4	BS_ACTION_CCRECP	Recovery record pointer
(C)	BITSTRING	1	BS_ACTION_FLAGS	Action Flags
	1...		BS_ACTION_ADD	0-delete 1-add
	.1..		BS_ACTION_CCWR	1-CC write/delete action
	..1.		BS_ACTION_CCDEL	1-CC action is delete
	...1		BS_ACTION_ CCONLY	1-CC action is delete
	.... 1...		BS_ACTION_CC	1 - A physical catalog I/O is required 0 - donot touch Log or cat
	.... .1..		BS_ACTION_ DELDONE	1 - node freemained
	.... ..1.		BS_ACTION_ COMMIT	1 - COMMIT_NOW on
	1		*	Reserved

Constants

Len	**Type**	**Value**	**Name**	**Description**
16	CHARACTER	>DFHBS_ACTION_BK	BS_ACTION_EYE	

Index

A

AB_CODE (1) BAACT 13
AB_CODE (10D) BAACT 18
AB_CODE (12D) BAACT 11
AB_PROGRAM (111) BAACT 18
AB_PROGRAM (131) BAACT 11
AB_PROGRAM (5) BAACT 13
ABCODE (180) APLI 4
ABEND (9EE) RMLK 308
ABEND (E6) RMLK 311
ABEND_AX_REGISTERS_ADDR (288) APLI 4
ABEND_FP_REGISTERS_ADDR (284) APLI 4
ABEND_GP_REGISTERS_ADDR (280) APLI 4
ABTERM_PENDING 1 DSTSK 67
ABTERM_PENDING_ECB (DC) DSTSK 66
ABYTE (0) FEP08 129
ACA 384
ACA_ACBP (50) TSAUX 384
ACA_ARROW (2) TSAUX 384
ACA_ASEGS 386
ACA_AUX_SPACE_QUEUE (28) TSAUX 384
ACA_BCAHA (88) TSAUX 385
ACA_BCAHD (84) TSAUX 385
ACA_BCAHF (8C) TSAUX 385
ACA_BCAP (140) TSAUX 386
ACA_BCID (110) TSAUX 386
ACA_BLKN (7C) TSAUX 385
ACA_BLOCK_NAME (8) TSAUX 384
ACA_BLOCK_NAME_STRING 8 TSAUX 389
ACA_BMLEN (134) TSAUX 386
ACA_BMP (120) TSAUX 386
ACA_BPSEG (118) TSAUX 386
ACA_BPSG2 (11C) TSAUX 386
ACA_BSEGS (146) TSAUX 386
ACA_BUFFER_QUEUE (38) TSAUX 384
ACA_BUWT (DC) TSAUX 385
ACA_BUWTH (E0) TSAUX 385
ACA_BWTN (D8) TSAUX 385
ACA_COMPARE_FAILED (BIT) TSAUX 386
ACA_COPIED_BMP (14C) TSAUX 386
ACA_CSA (104) TSAUX 386
ACA_CURWB (9A) TSAUX 385
ACA_DFH (3) TSAUX 384
ACA_DOMID (6) TSAUX 384
ACA_EXTEND_QUEUE (30) TSAUX 384
ACA_EXTENDING (BIT) TSAUX 386
ACA_FNCI (13C) TSAUX 386
ACA_FORMAT_BUFFERP (68) TSAUX 385
ACA_FORMAT_ECB (70) TSAUX 385
ACA_FORMAT_RBA (6C) TSAUX 385
ACA_FTIME (138) TSAUX 386
ACA_FULL 386
ACA_LAR (E4) TSAUX 385
ACA_LENGTH (0) TSAUX 384
ACA_MAPEP (128) TSAUX 386
ACA_MAPP (124) TSAUX 386
ACA_MAX_CIS_FORMATTED (64) TSAUX 385
ACA_MAXWB (98) TSAUX 385
ACA_MODEL_RPLP (60) TSAUX 385
ACA_NAG (D4) TSAUX 385
ACA_NAP (F0) TSAUX 386
ACA_NAVB (10C) TSAUX 386
ACA_NBCA (74) TSAUX 385
ACA_NCI (108) TSAUX 386
ACA_NCIA (BC) TSAUX 385
ACA_NCIAH (C0) TSAUX 385
ACA_NCOMP (F8) TSAUX 386
ACA_NIOER (FC) TSAUX 386
ACA_NP (E8) TSAUX 385
ACA_NPQ (EC) TSAUX 385
ACA_NSUSP (F4) TSAUX 386
ACA_NVCA (78) TSAUX 385
ACA_NVCAH (C4) TSAUX 385
ACA_OPENLIST_LENGTH (58) TSAUX 384
ACA_OPENLISTP (54) TSAUX 384
ACA_OPENSKELP (5C) TSAUX 385
ACA_PGCSA (100) TSAUX 386
ACA_PREFIX (0) TSAUX 384
ACA_RREFN (94) TSAUX 385
ACA_SPCI (114) TSAUX 386
ACA_SPCI1 386
ACA_SSP (12C) TSAUX 386
ACA_STATS (AC) TSAUX 385
ACA_STATS2 (E8) TSAUX 385
ACA_STRING_QUEUE (48) TSAUX 384
ACA_TRAP_FLAGS (148) TSAUX 386
ACA_TRDN (AC) TSAUX 385
ACA_TSBUFFER_SPTOKEN (20) TSAUX 384
ACA_TSS_SPTOKEN (18) TSAUX 384
ACA_TSX_SPTOKEN (10) TSAUX 384
ACA_TWTN (B0) TSAUX 385
ACA_TWTNF (B8) TSAUX 385
ACA_TWTNR (B4) TSAUX 385
ACA_VCAHD (90) TSAUX 385
ACA_VLKN (80) TSAUX 385
ACA_VUWT (CC) TSAUX 385
ACA_VUWTH (D0) TSAUX 385
ACA_VWTN (C8) TSAUX 385
ACA_WRITE_BUFFER_QUEUE (40) TSAUX 384
acb
  VTAM acb work area, FEP03 115
ACCEPT_PARMS (18) SOA 372
access
  data tables local access anchor blocks, DTCPS 68
ACCESS_CICS 4 SMDCC 363
ACCESS_ID (10) RMNM 321
ACCESS_ID (70) RMLK 317
ACCESS_ID (9CC) RMLK 308
ACCESS_ID (C4) RMLK 310
ACCESS_INVALID 4 SMDCC 363
ACCESS_READ_ONLY 4 SMDCC 363
ACCESS_USER 4 SMDCC 363
ACCESSIBLE (9ED) RMLK 308
ACCESSIBLE (E5) RMLK 311
ACT_ADD 5, 13, 14
ACT_COMPLETION_RESP (0) BAACT 14
ACT_GEN_NO (6C) BAACT 15
ACT_GEN_NO (DC) BAACT 16
ACT_IN_BUFFERS (BIT) BAACT 12, 18
ACT_INSTORE (BIT) BAACT 12, 18
ACT_KEY (0) BAACT 13
ACT_KEY (20) BAACT 5
ACT_KEY (4) BAACT 14
ACT_LR_KEY (3A) BAACT 15
ACT_LR_KEY (AA) BAACT 16
ACT_MODE (0) BAACT 14
ACT_NAME 5, 6, 7, 8, 10, 11, 12, 13, 14, 15, 16, 17, 19
ACT_REF (4) BAACT 14
ACT_REQ_PTR (78) BAACT 7, 8, 13, 19
action
  builder services action blocks, ZCQ 456
ACTIVATED 12, 18
ACTIVE (BIT) L2CH 220
activity
  bam activity class, BAACT 10
ACTIVITY (0) BAACT 10
ACTIVITY_ATTRIBS (0) BAACT 13
ACTIVITY_COMP_DATA (0) BAACT 13
ACTIVITY_ID (0) BAACT 14
ACTIVITY_RECORD (4) BAACT 12, 18
ACTIVITY_REF (0) BAACT 13
ACTIVITY_REQUEST (0) BAACT 15
ACTIVITY_SET (0) BAACT 13
ACTIVITY_SET_ELEMENT (0) BAACT 14
adapter
  adapter resource manager, FEP02 113
ADD_SUSPEND_ISSUED (BIT) DSTSK 66
affinities
  CICS affinities utility trace table, CAUTR 26
AIOCB_ADDR (1C) SOA 371
AIOCB_LEN (18) SOA 371
AKP_COUNT (BC) L2BS 213
AKP_COUNT (BC) L2SR 244
AKP_FREQUENCY 213, 244

AKP_FREQUENCY (18) L2SL 241
AKP_FREQUENCY (FC) L2CH 223
AKP_KICK_OFF 251
AKP_KICK_OFF 4 L2CH 224
AKP_MAX 4 L2SL 241
AKP_MIN 4 L2SL 241
AL_ACTIVITY 1 BAPT 24
AL_FULL 1 BAPT 24
AL_OFF 1 BAPT 24
AL_PROCESS 1 BAPT 24
ALL (BIT) STUCB 376
ALL_LINKS_CHAIN (18) RMLK 305
ALLOCATED (BIT) L2CH 220
ALREADY_AT_MAXOPEN (BIT) DSANC 58
ANC_ARROW (2) DMCB1 47
ANC_ARROW (2) LMCB1 204
ANC_ARROW (2) STCB1 374
ANC_ARROW (2) STUCB 375
ANC_BLOCK_NAME (8) DMCB1 47
ANC_BLOCK_NAME (8) LMCB1 204
ANC_BLOCK_NAME (8) STCB1 374
ANC_BLOCK_NAME (8) STUCB 375
ANC_DFH (3) DMCB1 47
ANC_DFH (3) LMCB1 204
ANC_DFH (3) STCB1 374
ANC_DFH (3) STUCB 375
ANC_DOMID (6) DMCB1 47
ANC_DOMID (6) LMCB1 204
ANC_DOMID (6) STCB1 374
ANC_DOMID (6) STUCB 375
ANC_FLAGS (3B) STCB1 374
ANC_FREECHAIN_1_GUARD (24) LMCB1 204
ANC_FREECHAIN_1_HEAD 204
ANC_FREECHAIN_1_NEXT (20) LMCB1 204
ANC_FREECHAIN_2_GUARD (2C) LMCB1 204
ANC_FREECHAIN_2_HEAD (28) LMCB1 204
ANC_FREECHAIN_2_NEXT (28) LMCB1 204
ANC_FREECHAIN_3_GUARD (34) LMCB1 204
ANC_FREECHAIN_3_HEAD (30) LMCB1 204
ANC_FREECHAIN_3_NEXT (30) LMCB1 204
ANC_LENGTH (0) DMCB1 47
ANC_LENGTH (0) LMCB1 204
ANC_LENGTH (0) STCB1 374
ANC_LENGTH (0) STUCB 375
ANC_MAXIMUM_TASKS (3C) LMCB1 205
ANC_NUMBER_OF_LOCKS (38) LMCB1 205
ANC_PREFIX (0) DMCB1 47
ANC_PREFIX (0) LMCB1 204
ANC_PREFIX (0) STCB1 374
ANC_PREFIX (0) STUCB 375
ANC_QUICKCELL_1_HEAD (10) LMCB1 204
ANC_QUICKCELL_2_HEAD (14) LMCB1 204
ANC_QUICKCELL_3_HEAD (18) LMCB1 204
ANC_SYSTEM_TERMINATING 374
ANC_SYSTEM_WAITS (70) DSANC 59
ANC_TASK_LIMIT (3C) LMCB1 205
ANC_TCB_DISP_TIME (68) DSANC 59
ANC_TCB_WAIT_TIME (60) DSANC 59
ANC_USER_EXIT_STATUS (BIT) STCB1 374
ANC_XTRA_LIMIT (3E) LMCB1 205
ANCH_ARROW (2) MEPS 257
ANCH_BLOCK_NAME (8) MEPS 257
ANCH_DFH (3) MEPS 257
ANCH_DOMID (6) MEPS 257
ANCH_LENGTH (0) MEPS 257
ANCH_PREFIX (0) MEPS 257
anchor
  data tables connection anchor blocks, DTLPS 69
  data tables local access anchor blocks, DTCPS 68
  data tables remote sharing anchor block, DTRPS 72
  data tables security anchor block, DTXPS 74
  data tables SVC routine anchor blocks, DTSPS 72
  dispatcher domain anchor block, DSANC 53
  document handler anchor block, DHANC 39
  domain manager anchor block, DMCB1 47
  enqueue domain anchor block, NQA 275
  kernel anchor block, KCB 151
  lock manager domain anchor block, LMCB1 204
  logger domain anchor block, LGANC 188
  message domain anchor block, MEPS 257
  parameter manager domain anchor block, PAA 283
  resource definition anchor block, RDAB 299
anchor *(continued)*
  security domain anchor block, XSANC 448
  sm macro-compatability anchor block, SMMCC 364
  sockets anchor block, SOA 366
  statistics domain anchor block, STCB1 374
  statistics utility program anchor block, STUCB 375
  storage manager anchor block, SMDCC 345
  temporary storage anchor block, TSA 380
  timer domain anchor block, TIA 378
  transaction manager domain anchor block, XMANC 435
  user domain anchor block, USANC 405
  web anchor block, WBABC 411
  web domain anchor block, WBANC 412
ANCHOR (0) DMCB1 47
ANCHOR (0) DSANC 53
ANCHOR (0) LMCB1 204
ANCHOR (0) MEPS 257
ANCHOR (0) STCB1 374
ANCHOR (0) STUCB 375
ANCHOR_ADDR (20) DSANC 61
AP (94) DSANC 55
APE (0) LDCBS 164
APE_ACTIVE 1 LDCBS 173
APE_AMODE_24 (BIT) LDCBS 165
APE_AMODE_31 165
APE_ANCHOR (FC) LDCBS 170
APE_ANCHOR_ID 8 LDCBS 173
APE_ARROW (2) LDCBS 164
APE_BLITO (84) LDCBS 165
APE_BLOCK_ID (8) LDCBS 164
APE_BUILT_BY_RESTART (BIT) LDCBS 165
APE_CELL_POOL_BDY 2 LDCBS 175
APE_CELL_POOL_NAME 8 LDCBS 174
APE_CHAIN_FIELDS (18) LDCBS 165
APE_CHAIN_SIZE (F4) LDCBS 170
APE_COPY_NUMBER (48) LDCBS 165
APE_CSECT_LIST_CHAIN_FIELDS (70) LDCBS 165
APE_CSECT_LIST_SIZE (6C) LDCBS 165
APE_CURRENT_USERS (58) LDCBS 165
APE_DFH (3) LDCBS 164
APE_DOMAIN (6) LDCBS 164
APE_DUMMY_CDE (80) LDCBS 165
APE_ENTRY_POINT (50) LDCBS 165
APE_FLAGS (45) LDCBS 165
APE_FREED 1 LDCBS 173
APE_ID_STRING 8 LDCBS 173
APE_LENGTH (0) LDCBS 164
APE_LOAD_POINT (4C) LDCBS 165
APE_LPA_LOADED (BIT) LDCBS 165
APE_MUSTDELET (BIT) LDCBS 165
APE_NEXT (18) LDCBS 165
APE_NIU_CHAIN_SIZE (F8) LDCBS 170
APE_OLDER_APE (20) LDCBS 165
APE_OLDER_APE_NIU (28) LDCBS 165
APE_ON_NIU_TIME (78) LDCBS 165
APE_OWNING_CPE (30) LDCBS 165
APE_PDB (34) LDCBS 165
APE_PREFIX (0) LDCBS 164
APE_PRIOR (1C) LDCBS 165
APE_PROGRAM_LENGTH (54) LDCBS 165
APE_PROGRAM_NAME (10) LDCBS 164
APE_RECOVERY_FLAGS (46) LDCBS 165
APE_REGION_LOADED (BIT) LDCBS 165
APE_RMODE_ANY (BIT) LDCBS 165
APE_RPL_LOADED (BIT) LDCBS 165
APE_STATUS (44) LDCBS 165
APE_STORAGE_SIZE (5C) LDCBS 165
APE_SUBPOOL_DATA (60) LDCBS 165
APE_YOUNGER_APE (24) LDCBS 165
APE_YOUNGER_APE_NIU (2C) LDCBS 165
APIQ 2
APIQ_ABEND 1 APIQ 3
APIQ_DISASTER 1 APIQ 3
APIQ_DPL_PROGRAM 1 APIQ 3
APIQ_DSA (38) APIQ 2
APIQ_DSA_X (BIT) APIQ 2
APIQ_EIB (1C) APIQ 2
APIQ_EIB_X (BIT) APIQ 2
APIQ_EXCEPTION 1 APIQ 3
APIQ_EXISTENCE 2
APIQ_FORMAT_NO 2
APIQ_FUNCTION (18) APIQ 2
APIQ_FUNCTION_X (BIT) APIQ 2

APIQ_HEAD (0) APIQ 2
APIQ_INQ_APPLICATION_DATA 1 APIQ 3
APIQ_INQ_FAILED 1 APIQ 3
APIQ_INVALID 1 APIQ 3
APIQ_INVALID_FUNCTION 1 APIQ 3
APIQ_KERNERROR 1 APIQ 3
APIQ_KERNHANDLE 2
APIQ_LOOP 1 APIQ 3
APIQ_NO_TRANSACTION_ENVIRONMENT 1 APIQ 3
APIQ_OK 1 APIQ 3
APIQ_PLISTLEN (0) APIQ 2
APIQ_PURGED 1 APIQ 3
APIQ_REASON (1B) APIQ 2
APIQ_REASON_X (BIT) APIQ 2
APIQ_RESPONSE 2
APIQ_RESPONSE_X 2
APIQ_RSA (34) APIQ 2
APIQ_RSA_X (BIT) APIQ 2
APIQ_SYSEIB (20) APIQ 2
APIQ_SYSEIB_X (BIT) APIQ 2
APIQ_TCTUA (24) APIQ 2
APIQ_TCTUA_X (BIT) APIQ 2
APIQ_TCTUASIZE (28) APIQ 2
APIQ_TCTUASIZE_X (BIT) APIQ 2
APIQ_TRANSACTION_DOMAIN_ERROR 1 APIQ 3
APIQ_TWA (2C) APIQ 2
APIQ_TWA_X (11) APIQ 2
APIQ_TWASIZE (30) APIQ 2
APIQ_TWASIZE_X (BIT) APIQ 2
APIQ_VERSION_NO (8) APIQ 2
APLI 3
application
  inquire application data xpi command, APIQ 2
APPLID (10) STUCB 375
APPLID (38) PAA 283
APPLID (44C) STUCB 375
APPLID_FLAGS (18) STUCB 375
APPLID_IGNORE (44C) STUCB 375
APPLID_SELECT (10) STUCB 375
APPLID_STATS (0) STUCB 377
APPLID_STATS_FOUND (BIT) STUCB 375
APPLID_STATS_PTR (810) STUCB 375
APPROX_SECONDS (0) FCQSE 106
area
  bind request save area, FEP04 116
  cics/db2 global work area, D2GWA 92
  common data area, FEP06 120
  conversation data area, FEP07 125
  cpi static storage area, CPSPS 34
  dump formatting communication area, DUFC 75
  enqueue domain queue element area, NQEA 277
  file browse work area for data tables, FBWAC 99
  language interface work area, APLI 3
  macro save area, PGA 285
  parameter area declarations, DUFP 76
  partner domain static storage area, PRS 296
  request parameter area, FEP17 141
  task browse area, DSTBA 63
  VTAM acb work area, FEP03 115
ARROW (182) DSANC 57
ARROW (2) CPSPS 34
ARROW (2) DSANC 54, 58, 60, 61, 62
ARROW (2) DSTBA 63
ARROW (2) PRS 296
ARROW (2) PTE 298
ARROW 1 DDCBC 37
ARROW 1 MEPS 259
ARROW 1 PAA 284
ARROW 1 SMDCC 356
ARROW 1 TIA 380
ARROW 1 TSA 381
ARROW 1 WBANC 413
ASYNCIO_PARMS (18) SOA 371
ATTRIBUTES (100) BAACT 11
ATTRIBUTES (E0) BAACT 18
ATTRIBUTES_PART (18) PTE 298
audit
  bam audit record class, BAAR 22
AUDIT_LEVEL (119) BAACT 18
AUDIT_LEVEL (139) BAACT 12
AUDIT_LEVEL (90) BAACT 6
AUDIT_LOG (11A) BAACT 18
AUDIT_LOG (13A) BAACT 12
AUDIT_LOG (91) BAACT 6
AUDITLEVEL (0) BAPT 23
authorised
  dm authorised facility state, DMAFC 45
  monitoring authorised parameter block, MNAFB 260
  statistics authorised parameter block, STAFB 373
AUTO_DELETE_FLAG (144) L2BS 216
AUTO_DELETE_FLAG (144) L2SR 248
AUTO_DELETE_FLAG (54) L2HS 230
AUTR_DISASTER 1 CAUTR 28
AUTR_EXCEPTION 1 CAUTR 28
AUTR_OK 1 CAUTR 28
AUTR_PURGED 1 CAUTR 28
auxiliary
  temporary storage auxiliary class, TSAUX 384
AVAIL (14) RMUW 335
AVERAGE_GAP (25C) L2BS 217
AVERAGE_GAP (25C) L2SR 249
AVL2 (0) DDBSC 35
AWAIT_CHAIN_FWD (C4) DSTSK 66
AWAITING_OPEN_TCB (66C) DSANC 58
AWAITING_OPEN_TCB_END (670) DSANC 58
AWAITING_OPEN_TCB_TOKEN (BC) DSTSK 66

B

BA_CATALOG_ERROR 4 BAPT 24
BA_DIRECTORY_ERROR 4 BAPT 24
BAAC_ACTIVITY_RECORD_TYPE 2 BAACT 20
BAAC_CLASS_DATA_TYPE (0) BAACT 16
BAAC_PERMANENT_STATE_TYPE (0) BAACT 17
BAAC_TRANSIENT_STATE_TYPE 18
BAACT 5, 9, 10, 21
BAAR 22
BABU_BUF_MODE (40) BAACT 12, 19
BABU_BUF_MODE (44) BAACT 7, 8
BABU_BUF_STATE (41) BAACT 12, 19
BABU_BUF_STATE (45) BAACT 7, 8
BABU_CURRENT_OFFS (54) BAACT 12, 19
BABU_CURRENT_OFFS (58) BAACT 7, 8
BABU_CURRENT_PTR (50) BAACT 12, 19
BABU_CURRENT_PTR (54) BAACT 7, 8
BABU_DUPLICATE 4 BAACT 20
BABU_FC_UTOKEN (6C) BAACT 12, 19
BABU_FC_UTOKEN (70) BAACT 7, 8
BABU_FILE_NOT_AUTH 4 BAACT 20
BABU_FILE_UNAVAILABLE 4 BAACT 20
BABU_FIRST_SEG (58) BAACT 12, 19
BABU_FIRST_SEG (5C) BAACT 7, 8
BABU_HEADER_LEN 4 BAACT 20
BABU_KEY_NOT_FOUND 4 BAACT 20
BABU_LOCKED 4 BAACT 20
BABU_MAX_SEG_LEN 4 BAACT 20
BABU_MODE_COPY 1 BAACT 20
BABU_MODE_DISK 1 BAACT 20
BABU_MODE_UNKN 1 BAACT 20
BABU_NEXT_SEG (58) BAACT 12, 19
BABU_NEXT_SEG (5C) BAACT 7, 8
BABU_PRIVATE 7, 8, 12, 19
BABU_PUBLIC (4) BAACT 12, 18
BABU_PUBLIC (8) BAACT 6, 7
BABU_READ_FAILURE 4 BAACT 20
BABU_REC_LEN (64) BAACT 12, 19
BABU_REC_LEN (68) BAACT 7, 8
BABU_RECORD_BUSY 4 BAACT 20
BABU_SEG_LEN 7, 8, 12, 19
BABU_SEG_LIST_HEAD (48) BAACT 12, 19
BABU_SEG_LIST_HEAD (4C) BAACT 7, 8
BABU_SEG_LIST_TAIL (4C) BAACT 12, 19
BABU_SEG_LIST_TAIL (50) BAACT 7, 8
BABU_SEQ (68) BAACT 12, 19
BABU_SEQ (6C) BAACT 7, 8
BABU_STATE_COPIED 1 BAACT 20
BABU_STATE_INIT 1 BAACT 19
BABU_STATE_NEW 1 BAACT 20
BABU_STATE_READ 1 BAACT 20
BABU_STATE_READING 1 BAACT 20
BABU_STATE_UNINIT 1 BAACT 19
BABU_STATE_WRITING 1 BAACT 20
BABU_STATE_WRITTEN 1 BAACT 20
BABU_STG_ADD (5C) BAACT 12, 19
BABU_STG_ADD (60) BAACT 7, 8

BABU_STG_LEN (60) BAACT   12, 19
BABU_STG_LEN (64) BAACT   7, 8
BABU_WRITE_FAILURE 4 BAACT   20
BABU_WRITE_STG_ADD (70) BAACT   12, 19
BABU_WRITE_STG_ADD (74) BAACT   7, 8
BACK_PTR   (20) DSANC   58
BACK_PTR   (4) DSANC   60
BACKLOG   (1C) SOA   371
BACKOUT_STATE (BIT) RMLK   315
BACKOUT_STATE (BIT) RMRO   325
BACKOUT_STATE (BIT) RMUW   334
BACKOUT_STRUCT (14) RMRO   325
BACKOUT_STRUCT (1C4) RMLK   315
BACKOUT_STRUCT (1C4) RMUW   334
BACKSTOP_STIMER_INDEX (1A) DSANC   61
BACKTRACK (C0) L2BS   213
BACKTRACK (C0) L2SR   244
BACO_CLASS_DATA_TYPE   21
BACO_FREE_SEGMENT   22
BACO_LENGTH_ERROR 4 BAACT   22
BACO_MAX_SEGMENT_LEN 4 BAACT   22
BACO_NEXT_SEGMENT (0) BAACT   22
BACO_SEGMENT_DATA (8) BAACT   22
BACO_SEGMENT_HEADER (0) BAACT   22
BACO_SEGMENT_LEN (4) BAACT   22
BACO_SEGMENT_TYPE   22
BACS_CONTAINER_NOT_FOUND 4 BAACT   10, 19
BACS_INVALID_CONTAINER_NAME 4 BAACT   10, 19
BACS_LENGTH_ERROR 4 BAACT   10, 19
BAD_EVENT (BIT) BAACT   16
BAEV_EYE_CATCHER (0) BAACT   10
BAEV_INSTANCE_DATA_BLOCK (0) BAACT   10
BALR_BROWSE_END 4 BAACT   19
BALR_DUPLICATE 4 BAACT   19
BALR_FILE_NOT_AUTH 4 BAACT   19
BALR_FILE_UNAVAILABLE 4 BAACT   19
BALR_FIRST_RECORD_NUMBER 4 BAACT   19
BALR_IO_ERROR 4 BAACT   19
BALR_LENGTH_ERROR 4 BAACT   19
BALR_LOCKED 4 BAACT   19
BALR_RECORD_NOT_FOUND 4 BAACT   19
BALR_TIMEOUT 4 BAACT   19
bam
    bam activity class, BAACT   10
    bam audit record class, BAAR   22
    bam contaier_set class, BAACT   9
    bam container class, BAACT   21
    bam process class, BAACT   5
    bam processtype class, BAPT   23
BAPR_EYE_CATCHER (0) BAACT   5
BAPR_PROCESS_INSTANCE_VER_1 2 BAACT   8
BAPR_PROCESS_RECORD_TYPE 2 BAACT   8
BAPR_TRANSIENT_STATE_TYPE (0) BAACT   7
BAPT   23
BAPT_CLASS_DATA_TYPE (0) BAPT   23
BATCH_CONTROL (10) DSANC   60
BATCH_CONTROL (190) DSANC   57
BATCH_CURRENT (14) DSANC   60
BATCH_CURRENT (194) DSANC   57
BATCH_REQD (BIT) DSTSK   66
BATCH_SIZE (10) DSANC   60
BATCH_SIZE (190) DSANC   57
BB   (2) TSAUX   388
BB   (2) TSMN   392
BBLX_ERROR_CODE 4 LGANC   193
BBLX_SIF_ERROR_CODE 4 LGANC   193
BC_ARROW   (2) DMCB2   49
BC_BLOCK_NAME (8) DMCB2   49
BC_CURSOR (10) DMCB2   49
BC_DFH   (3) DMCB2   49
BC_DOMID   (6) DMCB2   49
BC_LENGTH (0) DMCB2   49
BC_PREFIX (0) DMCB2   49
BCA   (0) TSAUX   387
BCA_BUFP   (C) TSAUX   387
BCA_CHNP   (4) TSAUX   387
BCA_CIB   (30) TSAUX   387
BCA_CIN   (14) TSAUX   387
BCA_FLAGS (2) TSAUX   387
BCA_LEN   (0) TSAUX   387
BCA_LOCK   (BIT) TSAUX   387
BCA_LR13   (20) TSAUX   387
BCA_NAP   387
BCA_NAPO   (0) TSAUX   387
BCA_NASP   (10) TSAUX   387
BCA_NFP   (8) TSAUX   387
BCA_NLP   (2C) TSAUX   387
BCA_RDN   (24) TSAUX   387
BCA_RECOV (BIT) TSAUX   387
BCA_RREFN (1C) TSAUX   387
BCA_TBW   (BIT) TSAUX   387
BCA_WBUF   (BIT) TSAUX   387
BCA_WCIB   (31) TSAUX   387
BCA_WCIN   (18) TSAUX   387
BCA_WTN   (28) TSAUX   387
BCI   387
BCI_CINR   (2) TSAUX   387
BCI_NASN   387
BCI_RDF   (4) TSAUX   387
BCI_RDFRE (7) TSAUX   388
BCI_RDFSG   388
BDY16   4 SMDCC   356
BDY16ROUND 4 SMDCC   356
BDY32   4 MEPS   259
BDY32   4 SMDCC   356
BDY32ROUND 4 SMDCC   356
BDY8   4 SMDCC   356
BFAC   (C) DDBSC   35
bind
    bind request save area, FEP04   116
BIND_PARMS (18) SOA   371
BIT_OFF   1 MEPS   259
BIT_ON   1 MEPS   259
BLANK   391
BLDL_ALIAS (BIT) LDCBS   166
BLDL_AMODE_31   166
BLDL_ARROW (4) LDCBS   165
BLDL_ATTRIBUTE   166
BLDL_BLOCK_ID (A) LDCBS   166
BLDL_C_FIELD (D) LDCBS   166
BLDL_DFH   (5) LDCBS   165
BLDL_DOMAIN (8) LDCBS   166
BLDL_ENTRIES (16) LDCBS   166
BLDL_ENTRY_POINT_OFFSET   166
BLDL_EXECUTABLE   166
BLDL_FLAGS_2   166
BLDL_ID_STRING 8 LDCBS   173
BLDL_LCN   (B) LDCBS   166
BLDL_LENGTH (0) LDCBS   165
BLDL_LENGTH_OF_ENTRY (14) LDCBS   166
BLDL_LIST (0) LDCBS   165
BLDL_LIST_ENTRY (0) LDCBS   166
BLDL_MACRO_PLIST (12) LDCBS   166
BLDL_NUMBER_IN_LIST (12) LDCBS   166
BLDL_PREFIX (0) LDCBS   165
BLDL_PROGRAM_LENGTH (18) LDCBS   166
BLDL_PROGRAM_NAME (0) LDCBS   166
BLDL_R   (A) LDCBS   166
BLDL_RMODE_ANY   166
BLDL_TT   (8) LDCBS   166
BLDL_TTRK (8) LDCBS   166
BLDL_WHERE_FOUND (C) LDCBS   166
BLK_NAME   (188) DSANC   57
BLK_NAME   (8) DSANC   54, 58, 60, 61, 62
block
    cics/db2 global block, D2GLB   85
    cics/db2 life of task block, D2LOT   93
    cpi-c conversation control block, CPCPS   32
    csub block, D2CSB   78
    data tables remote sharing anchor block, DTRPS   72
    data tables security anchor block, DTXPS   74
    db2entry block, D2ENT   81
    db2tran block, D2TRN   98
    dispatcher domain anchor block, DSANC   53
    document handler anchor block, DHANC   39
    domain manager anchor block, DMCB1   47
    enqueue domain anchor block, NQA   275
    file control cfdt uow pool block, FCUPC   107
    file control locks locator block, FLLBC   150
    kernel anchor block, KCB   151
    lock manager domain anchor block, LMCB1   204
    log manager block class, L2BL   208
    logger domain anchor block, LGANC   188
    message domain anchor block, MEPS   257
    monitoring authorised parameter block, MNAFB   260

block *(continued)*
  parameter manager domain anchor block, PAA 283
  resource definition anchor block, RDAB 299
  resource definition update block, RDUB 300
  security domain anchor block, XSANC 448
  session control request block, FEP18 145
  sm macro-compatability anchor block, SMMCC 364
  sockets anchor block, SOA 366
  statistics authorised parameter block, STAFB 373
  statistics domain anchor block, STCB1 374
  statistics utility program anchor block, STUCB 375
  storage manager anchor block, SMDCC 345
  temporary storage anchor block, TSA 380
  timer domain anchor block, TIA 378
  transaction manager domain anchor block, XMANC 435
  user domain anchor block, USANC 405
  user domain user data block, UDB 403
  VTAM receive request block, FEP15 139
  VTAM requests block, FEP16 140
  web anchor block, WBABC 411
  web domain anchor block, WBANC 412
  web request block class, WRB 427
BLOCK (0) L2BL 208
BLOCK (114) RMUW 339
BLOCK (474) RMLK 306
BLOCK (534) RMUW 339
BLOCK (54) RMLK 305
BLOCK_CONTEXT (D0) L2BS 213
BLOCK_CONTEXT (D0) L2SR 244
BLOCK_ID (20) L2BL 208
BLOCK_ID (60) L2CH 221
BLOCK_ID (8) L2HP 226
BLOCK_ID (98) L2CH 223
BLOCK_ID_USED (18) L2LF 235
BLOCK_ID_USED (18) LGSF 200
BLOCK_ID_USED (1C) L2LF 234
BLOCK_ID_USED (2C) L2LF 235, 236
BLOCK_ID_USED (2C) LGSF 200, 201
BLOCK_ID_USED (3C) L2LF 235
BLOCK_ID_USED (3C) LGSF 201
BLOCK_ID_USED (8) L2LF 231, 233
BLOCK_ID_USED (C) L2LF 234, 235
BLOCK_LENGTH (0) CPSPS 34
BLOCK_LENGTH (0) PRS 296
BLOCK_LENGTH (0) PTE 298
BLOCK_NAME (8) CPSPS 34
BLOCK_NAME (8) PRS 296
BLOCK_NAME (8) PTE 298
BLOCK_NUM (18) L2BL 208
BLOCK_OWNER (44) L2BS 212
BLOCK_OWNER (44) L2SR 244
BLOCK_OWNER (84) L2BS 213
BLOCK_OWNER (84) L2SR 244
BLOCK_OWNER (C) L2SR 250
BLOCK_PTR (0) L2BL 209
BLOCK_PTR (0) L2SR 250
BLOCK_PTR (18) L2CH 222
BLOCK_PTR (288) L2BS 217
BLOCK_PTR (2A0) L2BS 218
BLOCK_PTR (38) L2BS 212
BLOCK_PTR (38) L2SR 243
BLOCK_PTR (48) L2CH 221
BLOCK_PTR (78) L2BS 213
BLOCK_PTR (78) L2SR 244
BLOCK_PTR (8) L2RT 239
BLOCK_PTR (80) L2CH 221
BLOCKBUFFER (0) L2BL 209
BLOCKCONTEXT 209
BLOCKED (8D) BAACT 17
BLOCKED (AD) BAACT 11
BLOCKNAME_AH 8 DDCBC 38
BLOCKNAME_AN 8 DDCBC 38
BLOCKNAME_BV 8 DDCBC 38
BLOCKNAME_DDA 8 DDCBC 37
BLOCKNAME_DH 8 DDCBC 38
BLOCKNAME_HE 8 DDCBC 38
BLOCKNAME_HS 8 DDCBC 38
blocks
  bridge control blocks, BRDCC 24
  builder services action blocks, ZCQ 456
  data tables connection anchor blocks, DTLPS 69
  data tables local access anchor blocks, DTCPS 68
blocks *(continued)*
  data tables SVC routine anchor blocks, DTSPS 72
  directory manager building blocks, DDBSC 35
  external CICS interface control blocks, XCCBC 431
  kernel control blocks, KECB 155
  loader domain control blocks, LDCBS 164
  monitoring domain control blocks, MNCBS 262
  program manager control blocks, PGDCC 286
BLOCKS (118) RMUW 339
BLOCKS (478) RMLK 306
BLOCKS (538) RMUW 339
BLOCKS (58) RMLK 305
BLOCKSTATUS (0) L2SR 250
BMH (0) TSAUX 388
BMH_ARROW (4) TSAUX 388
BMH_BLOCK_NAME (A) TSAUX 388
BMH_BLOCK_NAME_STRING 6 TSAUX 389
BMH_DFH (5) TSAUX 388
BMH_DOMID (8) TSAUX 388
BMH_LENGTH (0) TSAUX 388
BMH_MAP_START (10) TSAUX 388
BMH_PREFIX (0) TSAUX 388
BMP (0) TSAUX 388
BODY (10) CPSPS 34
BODY (10) PRS 296
BODY (10) PTE 298
BPX_INTERFACE (0) SOA 371
BPX_LTE_PTR (14) SOA 371
BPX_PARAMETERS (18) SOA 371
BPX_REASON_CODE (8) SOA 371
BPX_RETURN_CODE (4) SOA 371
BPX_RETURN_VALUE (0) SOA 371
BPX_STE_PTR 371
BR_BFB_CATALOGUE_INTERVAL 4 BRDCC 26
BRACKET_FOUND (BIT) PAA 283
BRB (0) TSNM 394
BRB_CHANGE_COUNT (28) TSNM 394
BRB_NAME (18) TSNM 394
BRB_NEXT (0) TSNM 394
BRB_NODEP (2C) TSNM 394
BRB_PREV (4) TSNM 394
BRB_SLOTP (30) TSNM 394
BRB_TRANID (8) TSNM 394
BRB_TRANNUM (C) TSNM 394
BRB_TRANTOKEN (10) TSNM 394
BRDCC 24
bridge
  bridge control blocks, BRDCC 24
BRIDGE_FACILITY_TOKEN (10B) BAACT 16
BRIDGE_X (BIT) BAACT 16
BRLOGSTREAMTOKEN 218
BROKEN_LOG (1A9) L2BS 216
BROKEN_LOG (1A9) L2SR 248
BROKEN_LOG (B9) L2HS 230
BROKEN_RSN (1B0) L2BS 216
BROKEN_RSN (1B0) L2SR 248
BROKEN_RSN (C0) L2HS 230
BROKEN_RSP 216, 230, 248
browse 221
  domain manager browse cursor, DMCB2 49
  enqueue domain browse element, NQB 276
  enqueue domain browse owner extension, NQOX 279
  enqueue domain browse waiter extension, NQWX 282
  file browse work area for data tables, FBWAC 99
  task browse area, DSTBA 63
  transaction manager tran. browse element, XMXBC 441
BROWSE_ALL (BIT) L2CH 223
BROWSE_ALREADY_IN_PROGRESS 4 L2BS 219
BROWSE_AREA (0) DSTBA 63
BROWSE_CURSORS (0) DMCB2 49
BROWSE_ILLOGIC 4 L2CH 224
BROWSE_TOKENS 306
BROWSE_VAL (0) DDCBC 37
browseable
  log manager browseable stream class, L2BS 211
BROWSEABLE_STREAM_CLASS_DATA (0) L2BS 218
BROWSEABLE_STREAM_INSTANCE_DATA 217
BROWSEABLESTREAM (0) L2BS 211
BRPC (0) BRDCC 25
BRPC_BRDATA (40) BRDCC 25
BRPC_BRDATA_LEN (3C) BRDCC 25
BRPC_BREXIT_PROGRAM (18) BRDCC 25
BRPC_BRIDGE_TRANSACTION_ID (10) BRDCC 25

BRPC_EYE_CATCHER (4) BRDCC 25
BRPC_FLAGS (14) BRDCC 25
BRPC_LENGTH (0) BRDCC 25
BRPC_PREFIX (0) BRDCC 25
BRPC_TAKE_COPY (BIT) BRDCC 25
BRPC_USERID (20) BRDCC 25
BRPC_VERSION (C) BRDCC 25
BRSA (0) BRDCC 24
BRSA_BFB_INDEX 24
BRSA_BFB_KEEP_CHAIN (14) BRDCC 24
BRSA_BRPC_SUBPOOL (28) BRDCC 24
BRSA_BSB_SUBPOOL (30) BRDCC 24
BRSA_EYE_CATCHER (4) BRDCC 24
BRSA_GENERAL_SUBPOOL 24
BRSA_LENGTH (0) BRDCC 24
BRTA (0) BRDCC 24
BRTA_BFB_PTR (78) BRDCC 25
BRTA_BRDATA_LEN (84) BRDCC 25
BRTA_BRDATA_PTR (80) BRDCC 25
BRTA_BREXIT_INIT_OK (BIT) BRDCC 24
BRTA_BREXIT_PROGRAM (28) BRDCC 25
BRTA_BRIDGE_ENVIRONMENT 24
BRTA_BRIDGE_TRANSACTION_ID (10) BRDCC 24
BRTA_BRPC_LEN (94) BRDCC 25
BRTA_BRPC_PTR (90) BRDCC 25
BRTA_BRXA_LEN (8C) BRDCC 25
BRTA_BRXA_PTR (88) BRDCC 25
BRTA_CALL_EXIT_FOR_SYNC (15) BRDCC 24
BRTA_CONTEXT (14) BRDCC 24
BRTA_CONTEXT_BREXIT 1 BRDCC 26
BRTA_CONTEXT_BRIDGE 1 BRDCC 26
BRTA_CONTEXT_NORMAL 1 BRDCC 26
BRTA_CONTROL_BLOCKS (80) BRDCC 25
BRTA_EYE_CATCHER (4) BRDCC 24
BRTA_FACILITY (70) BRDCC 25
BRTA_FACILITY_TOKEN (70) BRDCC 25
BRTA_FLAGS (16) BRDCC 24
BRTA_FORMATTER_PROGRAM (30) BRDCC 25
BRTA_HEADER (0) BRDCC 24
BRTA_IDENTIFIER 25
BRTA_LENGTH (0) BRDCC 24
BRTA_LOAD_ADS_DESCRIPTOR (BIT) BRDCC 24
BRTA_NO 1 BRDCC 26
BRTA_ORIGINAL_NEXT_TRANID (7C) BRDCC 25
BRTA_START_CODE (18) BRDCC 25
BRTA_TRANSACTION_ID 25
BRTA_USERID (20) BRDCC 25
BRTA_YES 1 BRDCC 26
BRTOKEN_SUBPOOL (5F8) DSANC 57
BS_ACTION (0) ZCQ 456
BS_ACTION_ADD (BIT) ZCQ 457
BS_ACTION_ARRAY 457
BS_ACTION_CC (BIT) ZCQ 457
BS_ACTION_CCDEL (BIT) ZCQ 457
BS_ACTION_CCONLY (BIT) ZCQ 457
BS_ACTION_CCRECP (8) ZCQ 457
BS_ACTION_CCWR (BIT) ZCQ 457
BS_ACTION_COMMIT (BIT) ZCQ 457
BS_ACTION_CSIZE (26) ZCQ 457
BS_ACTION_DELDONE (BIT) ZCQ 457
BS_ACTION_ELEM (0) ZCQ 457
BS_ACTION_EYE 16 ZCQ 457
BS_ACTION_FLAGS (C) ZCQ 457
BS_ACTION_ID (8) ZCQ 456
BS_ACTION_MSIZE (24) ZCQ 457
BS_ACTION_NEXT (0) ZCQ 456
BS_ACTION_NODE (4) ZCQ 457
BS_ACTION_PATT (0) ZCQ 457
BS_ACTION_PLM (18) ZCQ 456
BS_ACTION_PREV (4) ZCQ 456
BS_ACTION_REQSTG (20) ZCQ 456
BS_BACKOUT_COMPLETE 0 RMRO 326
BS_BACKOUT_FAILED 0 RMRO 326
BS_NOT_BACKED_OUT 0 RMRO 326
BS_REBUILDING_FAILURE 0 RMRO 326
BS_RESET 0 RMRO 326
BSCD_CHAIN (10) L2BS 218
BSCD_EYE_CATCHER (0) L2BS 218
BSCD_FACTORY (38) L2BS 218
BSID_BACKTRACK 218
BSID_BROWSE_IN_PROGRESS (2B4) L2BS 218
BSID_CHAIN_HEAD (280) L2BS 217
BSID_CHAIN_LINK (270) L2BS 217
BSID_CHAINED (2B0) L2BS 218
BSID_EMPTY_STREAM (2B5) L2BS 218
BSID_EYE_CATCHER (260) L2BS 217
BSID_FLAGS 218
BSID_NEXT_RTOKEN 218
BTYPE (29) L2BL 208
BUF_APPENDS (234) L2BS 216
BUF_APPENDS (234) L2SR 248
BUF_FULL_WAITS (230) L2BS 216
BUF_FULL_WAITS (230) L2SR 248
BUFFER (38) L2BL 208
BUFFER_ARRAY_A 29
BUFFER_FULL 251
BUFFER_FULL 4 L2CH 224
BUFFER_LEN (160) L2BS 216
BUFFER_LEN (160) L2SR 248
BUFFER_LEN (70) L2HS 230
BUFFER_LENGTH 4 STUCB 378
BUFFER_LENGTH_ERROR 251
BUFFER_LENGTH_ERROR 4 L2CH 224
BUFFER_PTR (15C) L2BS 216
BUFFER_PTR (15C) L2SR 248
BUFFER_PTR (6C) L2HS 230
BUFFER_SIZE 2 PAA 284
BUILD_WAIT_LIST (BIT) DSANC 54
builder
  builder services action blocks, ZCQ 456
building
  directory manager building blocks, DDBSC 35
business
  web business logic compatibility interface, WBA1C 413
  web business logic interface parameters, WBBLC 416
BV_ARROW (2) DDCBC 37
BV_BLOCK_NAME (8) DDCBC 37
BV_DFH (3) DDCBC 37
BV_DOMID (6) DDCBC 37
BV_DONE_GETNEXT (BIT) DDCBC 37
BV_END (124) DDCBC 37
BV_FLAGS (20) DDCBC 37
BV_LENGTH (0) DDCBC 37
BV_NEXT (10) DDCBC 37
BV_OLDCURSOR (1C) DDCBC 37
BV_OLDDELETES (18) DDCBC 37
BV_OLDNAME 37
BV_ON_NAME (BIT) DDCBC 37
BV_PREFIX (0) DDCBC 37
BV_PREV (14) DDCBC 37
BV_TASK_RELATED (BIT) DDCBC 37
BYTE1 (2A8) APLI 5
BYTES_FOR_ABENDING_TASKS 4 SMDCC 356

C

C370_THREAD_TOKEN (10) APLI 3
C370_THREAD_WORKAREA_ADDR (1C) APLI 3
CAFF_EVENT_ENTRY (20) CAUTR 27
CAFF_EVENT_MODULE (0) CAUTR 27
CAFF_EVENT_MODULE (20) CAUTR 27
CAFF_EVENT_SPACE (25) CAUTR 27
CAFF_EVENT_SPACE (5) CAUTR 27
CAFF_EVENT_TEXT (26) CAUTR 27
CAFF_EVENT_TEXT (6) CAUTR 27
CAFF_EVENT_TIME (18) CAUTR 27
CAFF_EVENT_TIME (38) CAUTR 27
CAFFEVENTENTRY (0) CAUTR 27
CAFFEVENTTEXT (0) CAUTR 27
CALL_BACK_IN_PROGRESS (BIT) RMLK 307, 310
CANCEL_REQUEST 1 BAACT 20
CAT (0) SMDCC 356
CAT (0) TSA 381
CAT_ALIASES (68) XMCAT 438
CAT_BUFFERS (4) TSA 381
CAT_DSA_LIMIT 356
CAT_EDSA_LIMIT (8) SMDCC 356
CAT_EXTERNALS (0) XMCAT 438
CAT_FLAGS (0) SMDCC 356
CAT_FLAGS (0) TSA 381
CAT_MAX_ACTIVE (0) XMCAT 439
CAT_MXT_LIMIT (0) XMCAT 438
CAT_NAME 8 SMDCC 356
CAT_NAME 8 TSA 381
CAT_PURGE_THRESHOLD (4) XMCAT 439

CAT_START_COLD (BIT) TSA 381
CAT_STORAGE_PROTECT_REQ 356
CAT_STRINGS (8) TSA 381
CAT_TPNAME (78) XMCAT 438
CAT_TRAN_ISOLATION_REQ (BIT) SMDCC 356
CAT_TYPE 8 SMDCC 356
CAT_TYPE 8 TSA 381
CAT_TYPE_ME (16) CCGD 29
catalog
  catalog static storage, CCGD 29
  transaction manager catalog records, XMCAT 438
CATALOG_ACTIVE 29
CATALOG_ENTRY (0) PTE 299
CATALOG_ENTRY_NAME (0) PTE 299
CATALOG_RECORD (0) MEPS 258
CATALOG_RECORD (34) PAA 283
CATALOG_TYPE 29
catcher
  tsf - eye catcher map, FEP09 131
CATLG_TYPE 8 BAPT 24
CAUTR 26
CAUTRACE (0) CAUTR 26
CB_LENGTH (0) DSANC 53, 58, 60, 61, 62
CB_LENGTH (180) DSANC 57
CBYTE (0) FEP08 130
CC_ANC_ARROW (2) CCGD 29
CC_ANC_BLOCK_NAME (8) CCGD 29
CC_ANC_DFH (3) CCGD 29
CC_ANC_DOMID (6) CCGD 29
CC_ANC_EYECATCHER (2) CCGD 29
CC_LOCK 8 CCGD 31
CC_RECORD_FOR_PA (BIT) PAA 283
CC_SER_LOCK (3C) CCGD 29
CC_SER_LOCK_TOKEN (28) CCGD 29
CC_STARTUP_TASK (4C) CCGD 29
CC_STARTUP_TOKEN (48) CCGD 29
CC_STATIC_LEN (0) CCGD 29
CC_STRING_WAIT_ECB (2E) CCGD 29
CCANCHORB (0) CCGD 29
CCGD 29
CCSOPLMO 29
CDE_ID_STRING 8 LDCBS 174
CDSA 4 SMDCC 363
CDSA_NAME 5 LDCBS 175
CDSA_NAME 8 SMDCC 363
CE_ATTR_PART (0) PTE 299
CELINFO (268) APLI 4
CELINFO_HEAD (268) APLI 4
CELL_COUNT (B4) DSANC 55
CELL_COUNT (C4) DSANC 55
CELL_COUNT (D4) DSANC 55
CELL_COUNT (E4) DSANC 55
CELL_COUNT (F4) DSANC 55
CELL_FIELDS (0) DSTSK 67
CELL_HEADER (0) DSTBA 63
CELL_HEADER (0) DSTSK 64, 67
CELL_ID (10) DSTBA 63
CELL_ID (4) DSTSK 64, 67
CELL_PAGE_MAP (10) DSANC 62
CELL_TOKEN (4) DSTSK 64, 67
CEN_NAME_PART (0) PTE 299
cfdt
  file control cfdt pool element, FCPEC 101
  file control cfdt pool wait element, FCPWC 102
  file control cfdt uow pool block, FCUPC 107
CHAIN 6, 9, 11, 18
  log manager chain class, L2CH 219
CHAIN (0) L2CH 219
CHAIN_CLOSED (23) RMSL 327, 329
CHAIN_FACTORY (38) L2CH 222
CHAIN_FLAGS (40) DSTSK 65
CHAIN_FLAGS1 (40) DSTSK 65
CHAIN_FLAGS2 65
CHAIN_FLAGS3 (42) DSTSK 65
CHAIN_FLAGS4 (43) DSTSK 65
CHAIN_FREE_LIST 223
CHAIN_FREE_LIST_LINK (28) L2CH 220
CHAIN_HEADER (0) L2LF 233
CHAIN_INITIALISED (BIT) RMLK 314
CHAIN_INITIALISED (BIT) RMLS 319
CHAIN_INITIALISED (BIT) RMUW 333
CHAIN_LINK (0) BAACT 21
CHAIN_LINK (0) RMUW 335
CHAIN_LIST_LINK (18) L2CH 220
CHAIN_MANAGMENT (10) L2CH 222
CHAIN_PREV (24) L2LF 235, 236
CHAIN_PREV (24) LGSF 200, 201
CHAIN_PREV (4) L2LF 234, 235
CHAIN_PREV_DEAD (14) L2LF 234
CHAIN_PREV_DEAD (34) L2LF 235
CHAIN_PREV_DEAD (34) LGSF 201
CHAIN_PREV_LIVE (24) L2LF 235
CHAIN_PREV_LIVE (24) LGSF 200
CHAIN_PREV_LIVE (4) L2LF 234
CHAIN_PREV_SEC (24) L2LF 236
CHAIN_PREV_SEC (24) LGSF 201
CHAIN_PREV_SEC (4) L2LF 234
CHAIN_PTR (2C) RMUW 335
CHAIN_PTR (7C) L2CH 223
CHAINED (C1) L2BS 213
CHAINED (C1) L2SR 244
CHAINS_BROWSE_RESOURCES 223
CHAINS_ITER (68) L2CH 223
CHANGE_MODE_POSSIBLE (BIT) DSANC 57, 60
CHILD_MODE (0) BAACT 14
CHILDREN (94) BAACT 17
CHILDREN (B4) BAACT 11
CHOICE (53) RMLK 311
CHOICE (53) RMUW 331
CIB 387
CICS
  CICS affinities utility trace table, CAUTR 26
  external CICS interface control blocks, XCCBC 431
CICS 2 CCGD 31
CICS_RECORD_COUNT (8EC) STUCB 376
CICS_START_TIME 375
cics/db2
  cics/db2 global block, D2GLB 85
  cics/db2 global work area, D2GWA 92
  cics/db2 life of task block, D2LOT 93
  cics/db2 static storage, D2SS 96
CL_UH_END (C) L2LF 236
CL_UH_END (C) LGSF 203
CL_UH_JOURNAL_TYPE (4) L2LF 236
CL_UH_JOURNAL_TYPE (4) LGSF 203
CL_UH_LENGTH 203
CL_UH_LENGTH (0) L2LF 236
CL_UH_PREFIX_LENGTH (8) L2LF 236
CL_UH_PREFIX_LENGTH (8) LGSF 203
CL_UH_RSVD1 (6) L2LF 236
CL_UH_RSVD1 (6) LGSF 203
CL_USER_HEADER (0) LGSF 203
class
  bam activity class, BAACT 10
  bam audit record class, BAAR 22
  bam contaier_set class, BAACT 9
  bam container class, BAACT 21
  bam process class, BAACT 5
  bam processtype class, BAPT 23
  log manager block class, L2BL 208
  log manager browseable stream class, L2BS 211
  log manager chain class, L2CH 219
  log manager hard stream class, L2HS 227
  log manager history point class, L2HP 226
  log manager l2dm class, L2DM 224
  log manager lock tracker class, L2LT 238
  log manager record token class, L2RT 239
  log manager stream class, L2SR 242
  log manager system log class, L2SL 240
  logger reusable extended iliffe vector class, RUEI 343
  recovery manager link class data, RMLK 305
  recovery manager logname class data, RMNM 320
  recovery manager system log class data, RMSL 329
  recovery manager unit of work class data, RMUW 337
  sh request routing class, SHRTC 344
  temporary storage auxiliary class, TSAUX 384
  temporary storage main class, TSMN 392
  temporary storage model class, TSMN 390
  temporary storage name class, TSNM 393
  temporary storage ownership lock class, TSOL 394
  temporary storage queue class, TSQU 396
  temporary storage resource lock class, TSRL 401
  temporary storage shared class, TSRL 399
  temporary storage wait queue class, TSWQ 402
  transaction manager transaction class, XMCLC 439

class *(continued)*
  web request block class, WRB 427
CLASS_ACTIVE (BIT) XSSS 452, 453, 454
CLASS_BROWSE_RESOURCES (80) L2CH 223
CLASS_CHAIN (18) RMLK 309
CLASS_CHAIN (920) RMLK 307
CLASS_CMDSEC (BIT) XSSS 452, 453, 454
CLASS_DATA (0) RMNS 323
CLASS_DATA_BLOCK 305
CLASS_DATA_BLOCK (0) RMNM 320
CLASS_DATA_BLOCK (0) RMUW 337
CLASS_EYE_CATCHER (0) BAACT 16
CLASS_EYE_CATCHER (0) L2BL 209
CLASS_EYE_CATCHER (0) L2CH 222
CLASS_EYE_CATCHER (0) L2SR 249
CLASS_FLAGS (100) XSSS 453
CLASS_FLAGS (10A) XSSS 453
CLASS_FLAGS (114) XSSS 454
CLASS_FLAGS (11E) XSSS 454
CLASS_FLAGS (64) L2CH 223
CLASS_FLAGS (8) XSSS 454
CLASS_FLAGS (B0) XSSS 452
CLASS_FLAGS (BA) XSSS 453
CLASS_FLAGS (C4) XSSS 453
CLASS_FLAGS (CE) XSSS 453
CLASS_FLAGS (D8) XSSS 453
CLASS_FLAGS (E2) XSSS 453
CLASS_FLAGS (EC) XSSS 453
CLASS_FLAGS (F6) XSSS 453
CLASS_MEMBER_LENGTH (101) XSSS 453
CLASS_MEMBER_LENGTH (10B) XSSS 454
CLASS_MEMBER_LENGTH (115) XSSS 454
CLASS_MEMBER_LENGTH (11F) XSSS 454
CLASS_MEMBER_LENGTH (9) XSSS 454
CLASS_MEMBER_LENGTH (B1) XSSS 452
CLASS_MEMBER_LENGTH (BB) XSSS 453
CLASS_MEMBER_LENGTH (C5) XSSS 453
CLASS_MEMBER_LENGTH (CF) XSSS 453
CLASS_MEMBER_LENGTH (D9) XSSS 453
CLASS_MEMBER_LENGTH (E3) XSSS 453
CLASS_MEMBER_LENGTH (ED) XSSS 453
CLASS_MEMBER_LENGTH (F7) XSSS 453
CLASS_NAME (0) XSSS 454
CLASS_NAME (102) XSSS 453
CLASS_NAME (10C) XSSS 454
CLASS_NAME (116) XSSS 454
CLASS_NAME (A8) XSSS 452
CLASS_NAME (B2) XSSS 453
CLASS_NAME (BC) XSSS 453
CLASS_NAME (C6) XSSS 453
CLASS_NAME (D0) XSSS 453
CLASS_NAME (DA) XSSS 453
CLASS_NAME (E4) XSSS 453
CLASS_NAME (EE) XSSS 453
CLASS_NAME (F8) XSSS 453
CLASS_NAME 4 RMLK 309, 317
CLASS_PRIMARY_BROWSE (80) L2CH 223
CLASS_REBUILD 452, 453, 454
CLASS_RESSEC (BIT) XSSS 452, 453, 454
CLASS_SEC_BROWSE (BIT) L2CH 223
CLASS_SECONDARY_BROWSE (84) L2CH 223
CLASSDATABLOCK 26, 222, 240, 249
CLASSDATABLOCK (0) L2BL 209
CLASSENTRY (0) XSSS 454
CLIENT 315
CLIENT_IDENTITY_ADDRESS (1C) RMRO 325
CLIENT_IDENTITY_ADDRESS (1CC) RMLK 315
CLIENT_IDENTITY_ADDRESS (1CC) RMUW 334
CLIENT_NAME (0) RMLK 317
CLIENT_NAME (14) RMUW 335
CLIENT_NAME (54) RMLK 310
CLIENT_NAME (95C) RMLK 308
CLIENT_POINTER (40) RMLK 310
CLIENT_POINTER (948) RMLK 307
CLIENT_STATE (198) RMLK 314
CLIENT_STATE (198) RMUW 333
CLIENT_STATE_RECOVERED (BIT) RMLK 312
CLIENT_STATE_RECOVERED (BIT) RMUW 331
CLIENTID_ADDR (1C) SOA 372
CLIENTID_ADDR (20) SOA 372
CLIENTID_ADDR (24) SOA 372
CLIENTID_LENGTH (18) SOA 372
CLIENTID_LENGTH (1C) SOA 372
CLIENTID_LENGTH (20) SOA 372
CLOSE_PARMS (18) SOA 372
CLOSED 0 PAA 284
CM_COLLECT_OPTION (BIT) STCB1 374
CM_END_OF_DAY_TIME (20) STCB1 374
CM_END_OF_DAY_TOKEN (26) STCB1 374
CM_FLAGS (3A) STCB1 374
CM_INT_MICROSEC (14) STCB1 374
CM_INT_SEC (10) STCB1 374
CM_INTERVAL (10) STCB1 374
CM_INTERVAL_TOKEN (18) STCB1 374
CM_PEND_RESET_TIME (2E) STCB1 374
CM_PREV_RESET_TIME (34) STCB1 374
CM_USS_OPTION (BIT) STCB1 374
CMODE_COMPLETE 4 BAACT 20
CMODE_INITIAL 4 BAACT 20
CMODE_RUN 4 BAACT 20
COBOL2_CONTCODE (214) APLI 4
COBOL2_THREAD_TOKEN (0) APLI 3
COLD 1 PAA 284
COLD_START_CHAIN (50) RMSL 327, 329
COLL_APPLID (81A) STUCB 375
COLL_DATE (830) STUCB 375
COLL_JOBNAME (822) STUCB 375
COLL_LAST_RESET (928) STUCB 377
COLL_TIME (82A) STUCB 375
COLLECT_STATS (BIT) STUCB 376
COLLECTION_MANAGEMENT (10) STCB1 374
COMBO_SUBSPACE_OPEN_TYPES 4 SMDCC 363
COMBO_SUBSPACE_OPEN_TYPES 4 XMXDC 444
command
  inquire application data xpi command, APIQ 2
COMMIT_COMPLETE (BIT) RMLK 314
COMMIT_COMPLETE (BIT) RMUW 333
COMMIT_STATE (BIT) RMLK 315
COMMIT_STATE (BIT) RMRO 325
COMMIT_STATE (BIT) RMUW 334
COMMIT_STRUCT (18) RMRO 325
COMMIT_STRUCT (1C8) RMLK 315
COMMIT_STRUCT (1C8) RMUW 334
common
  common data area, FEP06 120
communication
  dump formatting communication area, DUFC 75
compatibility
  web business logic compatibility interface, WBA1C 413
COMPID 2 DDCBC 38
COMPID 2 LGANC 193
COMPID 2 SMDCC 362
COMPID 2 TSA 381
COMPID 2 USANC 406
COMPID 2 XSANC 449
COMPLETION_CODE (19) SOA 366
COMPLETION_CODE (1D) SOA 368
COMPLETION_CODE (24) DSTSK 64, 67
COMPLETION_CODE (3D) SOA 370
COMPLETION_CODE (41) SOA 370
COMPLETION_CODE (49) SOA 368
COMPLETION_CODE (75) SOA 367
COMPLETION_DATA (10C) BAACT 18
COMPLETION_DATA (12C) BAACT 11
COMPLETION_EVENT (11C) BAACT 11
COMPLETION_EVENT (1C) BAACT 13
COMPLETION_EVENT (FC) BAACT 18
COMPLETION_RESP (0) BAACT 13
COMPLETION_RESP (10C) BAACT 18
COMPLETION_RESP (12C) BAACT 11
COMPLETION_RESP_ABEND_R 1 BAACT 20
COMPLETION_RESP_FORCED 1 BAACT 20
COMPLETION_RESP_INCOMPLETE 1 BAACT 20
COMPLETION_RESP_NORMAL 1 BAACT 20
COMPONENT_ID 2 CCGD 31
COND 0 CCGD 31
CONNECT_FAILURE 251
CONNECT_FAILURE 4 L2HS 231
CONNECT_PARMS (150) XCCBC 433
CONNECTED (12B) L2BS 216
CONNECTED (12B) L2SR 248
CONNECTED (3B) L2HS 230
CONNECTED (C2) L2BS 213
CONNECTED (C2) L2SR 244
connection

connection *(continued)*
  connection descriptor, FEP05 117
  data tables connection anchor blocks, DTLPS 69
CONSOLE_FIRST_RECORD (BIT) PAA 283
CONSOLE_FLAG (BIT) PAA 283
constants
  web interface urp constants, WBUCC 424
contaier_set
  bam contaier_set class, BAACT 9
container
  bam container class, BAACT 21
CONTAINER (0) BAACT 21
CONTAINER_FLAGS (28) BAACT 21
CONTAINER_NAME (10) BAACT 21
CONTAINER_SET (0) BAACT 9
CONTAINERS (58) BAACT 5
CONTAINERS (A8) BAACT 18
CONTAINERS (C8) BAACT 11
CONTCODE (290) APLI 4
CONTCODE_BIT1 (BIT) APLI 4
CONTCODE_BIT2 (BIT) APLI 4
CONTINUE (100) RMLK 314
CONTINUE (100) RMUW 333
CONTINUE (48) RMLS 319
CONTINUE (A8) RMLK 313
CONTINUE (A8) RMUW 332
control
  bridge control blocks, BRDCC 24
  cpi-c conversation control block, CPCPS 32
  external CICS interface control blocks, XCCBC 431
  file control cfdt pool element, FCPEC 101
  file control cfdt pool wait element, FCPWC 102
  file control cfdt uow pool block, FCUPC 107
  file control locks locator block, FLLBC 150
  file control quiesce receive element, FCQRE 104
  file control quiesce send element, FCQSE 105
  kernel control blocks, KECB 155
  loader domain control blocks, LDCBS 164
  monitoring domain control blocks, MNCBS 262
  program manager control blocks, PGDCC 286
  session control request block, FEP18 145
CONTROL_POOL_BDY 2 LDCBS 175
CONTROL_POOL_NAME 8 LDCBS 174
CONTROL_POOL_NAME 8 MNCBS 273
conversation
  conversation data area, FEP07 125
  cpi-c conversation control block, CPCPS 32
CONVERSATION_ID (10) CPCPS 32
CONVERSATION_STATE (B8) CPCPS 33
CONVERSATION_TYPE (20) CPCPS 32
CONVERT (0) WRB 430
CONVERT_NO 1 WRB 430
CONVERT_YES 1 WRB 430
COORDINATOR (15) RMLK 317
COORDINATOR (40) RMLS 319
COORDINATOR (69) RMLK 310
COORDINATOR (971) RMLK 308
COORDINATOR (A0) RMLK 313
COORDINATOR (A0) RMUW 332
COORDINATOR (F8) RMLK 313
COORDINATOR (F8) RMUW 333
COUNT 223
COUNTS (648) DSANC 58
CPC_EYECATCHER (2) CPCPS 32
CPC_RECORD_LENGTH (0) CPCPS 32
 CPCPS 32
CPE (0) LDCBS 166
CPE_AMODE_31 167
CPE_ANCHOR (DC) LDCBS 170
CPE_ANCHOR_ID 8 LDCBS 173
CPE_APE_ANCHOR_ID 8 LDCBS 173
CPE_APE_CHAIN_FIELDS (78) LDCBS 168
CPE_APE_CHAIN_SIZE (74) LDCBS 168
CPE_APE_CREATING 1 LDCBS 174
CPE_ARROW (2) LDCBS 166
CPE_ATTRIBUTES 167
CPE_BAD 1 LDCBS 173
CPE_BLITO (AC) LDCBS 168
CPE_BLOCK_ID (8) LDCBS 166
CPE_BUILT_BY_RESTART (BIT) LDCBS 167
CPE_C_BYTE (29) LDCBS 167
CPE_CC_DONE 1 LDCBS 173
CPE_CC_REQD 1 LDCBS 174
CPE_CELL_POOL_BDY 2 LDCBS 175
CPE_CELL_POOL_NAME 8 LDCBS 174
CPE_CHAIN_SIZE (D8) LDCBS 170
CPE_COMPRESSIONS (9C) LDCBS 168
CPE_CSECTL_CREATING 1 LDCBS 174
CPE_CURRENT_USERS (6C) LDCBS 168
CPE_DE (1C) LDCBS 167
CPE_DELETED 1 LDCBS 173
CPE_DFH (3) LDCBS 166
CPE_DISCONNECTING 1 LDCBS 174
CPE_DOMAIN (6) LDCBS 166
CPE_ENTRY_POINT_OFFSET 167
CPE_EYE_CATCH (6) LDCBS 166
CPE_EYE_CATCH_I 6 LDCBS 174
CPE_FETCH_COUNT (94) LDCBS 168
CPE_FLAGS 167
CPE_FREED 1 LDCBS 173
CPE_GLOB_PTR (A8) LDCBS 168
CPE_ID_STRING 8 LDCBS 173
CPE_LCN (27) LDCBS 167
CPE_LENGTH (0) LDCBS 166
CPE_LOAD_COUNT (70) LDCBS 168
CPE_LOAD_TIME (98) LDCBS 168
CPE_LOADED 1 LDCBS 173
CPE_LOADED_BY_RESTART (BIT) LDCBS 167
CPE_LOCATED 1 LDCBS 173
CPE_LOCK (19) LDCBS 167
CPE_LPA_LOCATING 1 LDCBS 174
CPE_MUSTDELET (BIT) LDCBS 167
CPE_NEXT (10) LDCBS 166
CPE_OLD_COPY_IN_LPA (BIT) LDCBS 167
CPE_PDB (58) LDCBS 168
CPE_PDB_CATALOG_STATUS 167
CPE_PMARL_VALID (BIT) LDCBS 167
CPE_PREFIX (0) LDCBS 166
CPE_PRIOR (14) LDCBS 166
CPE_PROGRAM_ACQUIRED (BIT) LDCBS 167
CPE_PROGRAM_LENGTH (34) LDCBS 167
CPE_PROGRAM_NAME (1C) LDCBS 167
CPE_PROGRAM_STATUS (18) LDCBS 167
CPE_PRVMOD (BIT) LDCBS 167
CPE_R (26) LDCBS 167
CPE_RECOVERY_FLAGS (1A) LDCBS 167
CPE_REENTRANT (BIT) LDCBS 167
CPE_REFRESHES (A4) LDCBS 168
CPE_RMODE_ANY 167
CPE_RPL_LOADING 1 LDCBS 174
CPE_RPL_LOCATING 1 LDCBS 174
CPE_STATS (90) LDCBS 168
CPE_TIMES_USED (90) LDCBS 168
CPE_TT (24) LDCBS 167
CPE_TTRK (24) LDCBS 167
CPE_UNLOCKED 1 LDCBS 174
CPE_UNUSED 1 LDCBS 173
CPE_USES (68) LDCBS 168
CPE_WAITS (A0) LDCBS 168
CPE_Z_BYTE (28) LDCBS 167
cpi
  cpi static storage area, CPSPS 34
cpi-c
  cpi-c conversation control block, CPCPS 32
CPI_ACQUIRE_SUSPEND_TOK_FAILED 2 CPSPS 35
CPI_ACQUIRED_SUSPEND_TOK 2 CPSPS 35
CPI_INIT_SUCCEEDED 2 CPSPS 35
CPI_INIT_TASK_ATTACHED 2 CPSPS 35
CPI_INIT_TASK_STARTED 2 CPSPS 35
CPI_LOAD_CPIC_FAILED 2 CPSPS 35
CPI_LOAD_CPIRR_FAILED 2 CPSPS 35
CPI_LOADED_CPIC 2 CPSPS 35
CPI_LOADED_CPIRR 2 CPSPS 35
CPI_OPEN_FOR_BUSINESS 2 CPSPS 35
CPI_SSA (0) CPSPS 34
CPI_SSA_BLOCK_NAMEI 8 CPSPS 35
CPI_SSA_LENGTH 1 CPSPS 35
CPI_STATIC_STORAGE_INITIALIZED 2 CPSPS 35
CPIC_LAST_CONVID 34
CPIC_LOG_DATA (0) CPCPS 33
 CPSPS 34
CS_BUILDING_TBF 0 RMRO 326
CS_COMMIT_COMPLETE 0 RMRO 326
CS_COMMIT_FAILED 0 RMRO 326
CS_GROUP (44) DSTSK 65
CS_OFFSET (60) BAACT 6

CS_OFFSET (8) BAACT 9
CS_OFFSET (B0) BAACT 18
CS_OFFSET (D0) BAACT 11
CS_RESET 0 RMRO 326
CSA_ADDRESS (94) DSANC 55
CSB_ACCOUNT_CLOCK (9C) D2CSB 79
CSB_ACCOUNT_LUNAME (94) D2CSB 79
CSB_ACCOUNT_NETNAME (8C) D2CSB 79
CSB_ACCOUNT_TOKEN (8C) D2CSB 79
CSB_ACCOUNT_TOKEN_ACTIVE (BIT) D2CSB 79
CSB_ACCOUNT_TOKEN_FLAG (A2) D2CSB 79
CSB_ACEE_ADDRESS (80) D2CSB 79
CSB_ACTIVE_NEXT (34) D2CSB 78
CSB_ACTIVE_PREV (30) D2CSB 78
CSB_ATTACH_DETACH_NEXT (58) D2CSB 79
CSB_ATTACH_TASK (BIT) D2CSB 79
CSB_AVAIL_ASSIGN (BIT) D2CSB 79
CSB_CHAP (A5) D2CSB 79
CSB_CLOCK (10) D2CSB 78
CSB_CORRELATION_ID (74) D2CSB 79
CSB_CTL1 79
CSB_CTL2 79
CSB_CURRENT_TRACE_ENTRY (1CC) D2CSB 80
CSB_CURSOR (BIT) D2CSB 79
CSB_DETACH_TASK (BIT) D2CSB 79
CSB_ECB (28) D2CSB 78
CSB_ERROR_BUFFER (194) D2CSB 80
CSB_EYE (2) D2CSB 78
CSB_FRB (CC) D2CSB 80
CSB_GLB_ADDRESS (18) D2CSB 78
CSB_GLB_PTHREAD_NEXT (44) D2CSB 79
CSB_GLB_PTHREAD_PREV (40) D2CSB 79
CSB_GLB_TCB_NEXT (54) D2CSB 79
CSB_GLB_TCB_PREV (50) D2CSB 79
CSB_INITIAL_STATE (BIT) D2CSB 79
CSB_LENGTH (0) D2CSB 78
CSB_LOT_ADDRESS (20) D2CSB 78
CSB_NETWORK_ID (B8) D2CSB 80
CSB_PLAN_NAME (5C) D2CSB 79
CSB_PREFIX (0) D2CSB 78
CSB_PRIMARY_AUTH_NAME (64) D2CSB 79
CSB_PRIMARY_AUTH_SAVEAREA (A8) D2CSB 80
CSB_PROTECTED_THREAD (BIT) D2CSB 79
CSB_RCT_ADDRESS (1C) D2CSB 78
CSB_RCT_PTHREAD_NEXT (3C) D2CSB 79
CSB_RCT_PTHREAD_PREV (38) D2CSB 79
CSB_RCT_TCB_NEXT (4C) D2CSB 79
CSB_RCT_TCB_PREV (48) D2CSB 79
CSB_REQUEST_NUMBER (1C8) D2CSB 80
CSB_SAVEAREA (FC) D2CSB 80
CSB_SDWA_ADDRESS (2E0) D2CSB 80
CSB_SDWA_NAME (2D8) D2CSB 80
CSB_SDWA_PSW (2D0) D2CSB 80
CSB_SDWA_REGS (290) D2CSB 80
CSB_SECONDARY_AUTH_NAME (6C) D2CSB 79
CSB_SECONDARY_AUTH_SAVEAREA (B0) D2CSB 80
CSB_SIGNON_TIME (84) D2CSB 79
CSB_SUBTASK_IN_DB2 (BIT) D2CSB 79
CSB_SUBTASK_RUNNING (BIT) D2CSB 79
CSB_TASK_ATTACHED_OK (BIT) D2CSB 79
CSB_TASK_TERMED_ABNORMAL (BIT) D2CSB 79
CSB_TASK_TERMED_OK (BIT) D2CSB 79
CSB_TCB_ADDRESS (24) D2CSB 78
CSB_TERM_THREAD (BIT) D2CSB 79
CSB_TERMINATE_ECB (2C) D2CSB 78
CSB_TERMINATE_TASK (BIT) D2CSB 79
CSB_THREAD_CREATED (BIT) D2CSB 79
CSB_THREAD_NUMBER (A6) D2CSB 79
CSB_THREAD_NUMBER_DEC (7C) D2CSB 79
CSB_TRACE_ENTRIES_START (1E0) D2CSB 80
CSB_TRACE_FRBRC1 80
CSB_TRACE_FRBRC2 (C) D2CSB 80
CSB_TRACE_HEAD (1D0) D2CSB 80
CSB_TRACE_HEAD_EYE 16 D2CSB 81
CSB_TRACE_REQUEST (4) D2CSB 80
CSB_TRACE_REQUEST_NUM (0) D2CSB 80
CSB_TRACE_TABLE_ENTRY (1E0) D2CSB 80
CSB_TRACE_TAIL (280) D2CSB 80
CSB_TRACE_TAIL_EYE 16 D2CSB 81
CSB_TRANSID (78) D2CSB 79
CSB_TYPE (74) D2CSB 79
CSB_WLM_PERF_TOKEN (C8) D2CSB 80
CSB_WORKAREA (144) D2CSB 80
CSECTL (0) LDCBS 168
CSECTL_ADDRESS (8) LDCBS 168
CSECTL_ARROW (2) LDCBS 168
CSECTL_BLOCK_ID (8) LDCBS 168
CSECTL_CELL_POOL_BDY 2 LDCBS 175
CSECTL_CELL_POOL_NAME 8 LDCBS 174
CSECTL_CHAIN_FIELDS (10) LDCBS 168
CSECTL_CICS_VERSION (C) LDCBS 168
CSECTL_CREATION (18) LDCBS 168
CSECTL_DFH (3) LDCBS 168
CSECTL_DOMAIN (6) LDCBS 168
CSECTL_ENTRIES (18) LDCBS 168
CSECTL_ENTRY (0) LDCBS 168
CSECTL_ID_STRING 8 LDCBS 174
CSECTL_LENGTH (0) LDCBS 168
CSECTL_MODULE (0) LDCBS 168
CSECTL_NEXT (10) LDCBS 168
CSECTL_NUMBER_OF_ENTRIES 1 LDCBS 174
CSECTL_PREFIX (0) LDCBS 168
CSECTL_PRIOR (14) LDCBS 168
CSECTL_PTF_LEVEL (10) LDCBS 168
CSQC_ERROR_CODE 4 LGANC 193
CSTP_AREA (A0) DSANC 55
CSTP_ECB_LIST (A4) DSANC 55
CSTP_FLAGS (A8) DSANC 55
CSTP_MUST_DSP (BIT) DSANC 55
CSTP_TASK_REF (A0) DSANC 55
CSTP_WAITING (BIT) DSANC 55
csub
    csub block, D2CSB 78
CTL 387
CTL_NAME (0) TSAUX 387
CTL_NAME_STRING 8 TSAUX 389
CTN (0) SMDCC 350
CTN_ADDR (8) SMDCC 350
CTN_LEFT (0) SMDCC 350
CTN_LEN (C) SMDCC 350
CTN_PPXP (10) SMDCC 350
CTN_RIGHT (4) SMDCC 350
CTNBLOCK_SIZE 4 SMDCC 363
CTSD_ATTACH_PARMS (0) TSA 381
CTSD_LASTREF_TIME (10) TSA 381
CTSD_QUEUE_NAME (0) TSA 381
CUR_BLOCK_ID (190) L2BS 216
CUR_BLOCK_ID (190) L2SR 248
CUR_BLOCK_ID (A0) L2HS 230
CUR_TIME_GMT (198) L2BS 216
CUR_TIME_GMT (198) L2SR 248
CUR_TIME_GMT (A8) L2HS 230
CUR_TIME_LOCAL (1A0) L2BS 216
CUR_TIME_LOCAL (1A0) L2SR 248
CUR_TIME_LOCAL (B0) L2HS 230
CUR_TIMESTAMP (198) L2BS 216
CUR_TIMESTAMP (198) L2SR 248
CUR_TIMESTAMP (A8) L2HS 230
CURR_ALLOC_OPEN_TCBS (648) DSANC 58
CURR_BLOCK_NUM (0) L2BL 209
CURR_BLOCK_NUM (D0) L2BS 213
CURR_BLOCK_NUM (D0) L2SR 244
CURR_OPEN_TCBS (650) DSANC 58
CURRENT (30) L2BS 212
CURRENT (30) L2SR 243
CURRENT (40) L2BL 208
CURRENT (8) L2BL 209
CURRENT_APPLID (83E) STUCB 376
CURRENT_CHAIN_PTR (88) L2CH 223
CURRENT_CICS_START_TIME (89C) STUCB 376
CURRENT_DATE (84C) STUCB 376
CURRENT_ENTRY_POINT 376
CURRENT_HP (90) L2CH 223
CURRENT_INTERVAL (848) STUCB 376
CURRENT_INTERVAL_TIME (8F9) STUCB 376
CURRENT_LINK_PTR (48) RMLK 310
CURRENT_LINK_PTR (950) RMLK 307
CURRENT_NUM_APPLID (868) STUCB 376
CURRENT_PARM_LIST (7C) DSTSK 66
CURRENT_PASS_NUMBER 376
CURRENT_POOL (44) PAA 283
CURRENT_POSITION (10) CAUTR 27
CURRENT_REC (4C) PAA 283
CURRENT_RECORD_TYPE (892) STUCB 376
CURRENT_REPORT_TYPE (862) STUCB 376
CURRENT_REQ_TOKEN (85A) STUCB 376

CURRENT_REQUEST (77) DSTSK 66
CURRENT_RESOURCE_ID (86A) STUCB 376
CURRENT_STORAGE_FREE (74) DSANC 54
CURRENT_STREAM (38) L2CH 220
CURRENT_TCB_DATA (78) DSTSK 66
CURRENT_TIME (120) DSANC 56
CURRENT_TIME (854) STUCB 376
CURRENT_VERSION (8F8) STUCB 376
CURRNODE (28) RMUW 335
CURRNODE (78) L2CH 223
cursor
  domain manager browse cursor, DMCB2 49

D

D2CSB 78
D2ENT 81
D2GLB 85
D2GWA 92
D2LOT 93
D2S_ATHREAD_LOCK_TOKEN (38) D2SS 97
D2S_D2CSB_DIR_TOKEN (20) D2SS 96
D2S_D2CSB_SM_TOKEN (50) D2SS 97
D2S_D2ENT_DIR_TOKEN (14) D2SS 96
D2S_D2ENT_LOCK_TOKEN (28) D2SS 96
D2S_D2ENT_SM_TOKEN (40) D2SS 97
D2S_D2GLB_LOCK_TOKEN (24) D2SS 96
D2S_D2ST_DISASTER 1 D2SS 97
D2S_D2ST_EXCEPTION 1 D2SS 97
D2S_D2ST_OK 1 D2SS 97
D2S_D2ST_RESP (72) D2SS 97
D2S_D2TRN_LOCK_TOKEN (2C) D2SS 96
D2S_D2TRN_N_DIR_TOKEN (18) D2SS 96
D2S_D2TRN_SM_TOKEN (48) D2SS 97
D2S_D2TRN_T_DIR_TOKEN (1C) D2SS 96
D2S_DB2ENTRY_CHANGE_COUNT (68) D2SS 97
D2S_DB2TRAN_CHANGE_COUNT (6C) D2SS 97
D2S_DFHD2CC_ENTRY_POINT (58) D2SS 97
D2S_DFHD2GLB (10) D2SS 96
D2S_DFHD2STP_ENTRY_POINT (60) D2SS 97
D2S_DFHD2STR_ENTRY_POINT (5C) D2SS 97
D2S_DFHD2TM_ENTRY_POINT (64) D2SS 97
D2S_DISCONNECT_ECB 97
D2S_EYE (2) D2SS 96
D2S_FREE_TCB_LOCK_TOKEN (30) D2SS 96
D2S_INIT_ECB (70) D2SS 97
D2S_INIT_ECB_POSTED 97
D2S_LENGTH (0) D2SS 96
D2S_LOT_LOCK_TOKEN (3C) D2SS 97
D2S_PREFIX (0) D2SS 96
D2S_PTHREAD_LOCK_TOKEN (34) D2SS 97
D2SS 96
D2TRN 98
DAILY 0 STUCB 378
DASD_ONLY_FLAG (12D) L2BS 216
DASD_ONLY_FLAG (12D) L2SR 248
DASD_ONLY_FLAG (3D) L2HS 230
data
  common data area, FEP06 120
  conversation data area, FEP07 125
  data tables connection anchor blocks, DTLPS 69
  data tables local access anchor blocks, DTCPS 68
  data tables remote sharing anchor block, DTRPS 72
  data tables security anchor block, DTXPS 74
  data tables SVC routine anchor blocks, DTSPS 72
  file browse work area for data tables, FBWAC 99
  inquire application data xpi command, APIQ 2
  recovery manager link class data, RMLK 305
  recovery manager logname class data, RMNM 320
  recovery manager system log class data, RMSL 329
  recovery manager unit of work class data, RMUW 337
  security domain transaction data, XSXD 455
  user domain transaction data, USXD 410
  user domain user data block, UDB 403
  web state manager data, WBSTC 422
DATA (10) DDBSC 35
DATA (7C) RMDM 301
DATA (80) L2DM 224
DATA_ADDRESS (24) BAACT 21
DATA_LENGTH (20) BAACT 21
DATA_NOT_FOUND 251
DATA_NOT_FOUND 4 L2BL 211
db2entry
  db2entry block, D2ENT 81
db2tran
  db2tran block, D2TRN 98
DBB (0) DHANC 41
DBB_ARROW (2) DHANC 41
DBB_BKMARK_NAME (24) DHANC 41
DBB_BLOCK_NAME (8) DHANC 41
DBB_BOOKMARK (BIT) DHANC 41
DBB_DFH (3) DHANC 41
DBB_DOMID (6) DHANC 41
DBB_LENGTH (0) DHANC 41
DBB_NEXT_BKMARK (1C) DHANC 41
DBB_NEXT_CELEM (10) DHANC 41
DBB_PREFIX (0) DHANC 41
DBB_PREV_BKMARK (20) DHANC 41
DBB_PREV_CELEM (14) DHANC 41
DCD_ABEND 8 LGANC 192
DCD_ABEND 8 SMDCC 362
DCD_ABEND 8 TSA 381
DCD_ABEND 8 USANC 406
DCD_ABEND 8 XSANC 449
DCD_APPCLU_RACLIST_FAILED 8 XSANC 449
DCD_FAQE_ERROR 8 SMDCC 362
DCD_INCOMPLETE_UOW_ERROR 8 RMUW 336, 341
DCD_LOOP 8 SMDCC 362
DCD_LOOP 8 USANC 406
DCD_LOOP 8 XSANC 449
DCD_NO_MVS_STORAGE 8 SMDCC 362
DCD_NO_MVS_STORAGE 8 USANC 406
DCD_NO_MVS_STORAGE 8 XSANC 449
DCD_NO_STORAGE 8 LGANC 192
DCD_NO_STORAGE 8 SMDCC 362
DCD_NO_STORAGE 8 USANC 406
DCD_NO_STORAGE 8 XSANC 449
DCD_SEVERE_ERROR 8 LGANC 192
DCD_SEVERE_ERROR 8 SMDCC 362
DCD_SEVERE_ERROR 8 TSA 381
DCD_SEVERE_ERROR 8 USANC 406
DCD_SEVERE_ERROR 8 XSANC 449
DCD_STCK_ERROR 8 SMDCC 362
DCD_STCK_ERROR 8 USANC 406
DCD_STCK_ERROR 8 XSANC 449
DCD_STORAGE_VIOLATION 8 SMDCC 362
DCD_STREAM_DEFINE_ERROR 8 LGANC 192
dce
  dce services domain global statistics, DEGPC 38
DCHAIN (10) RMNS 322
DCHAINNODE (0) RMNS 322
DCR (0) DHANC 40
DCR_ARROW (2) DHANC 40
DCR_BLOCK_NAME (8) DHANC 40
DCR_DATA_SIZE (3C) DHANC 40
DCR_DFH (3) DHANC 40
DCR_DOCUMENT_COUNT (28) DHANC 40
DCR_DOCUMENT_SIZE (2C) DHANC 40
DCR_DOMID (6) DHANC 40
DCR_EMBED_DEPTH (58) DHANC 40
DCR_FIRST_CELEM (18) DHANC 40
DCR_FIRST_DBP (20) DHANC 40
DCR_FIRST_TEMPLATE (50) DHANC 40
DCR_LAST_CELEM (1C) DHANC 40
DCR_LAST_DBP (24) DHANC 40
DCR_LAST_TEMPLATE (54) DHANC 40
DCR_LENGTH (0) DHANC 40
DCR_NEXT (10) DHANC 40
DCR_NUM_BKMARKS (30) DHANC 40
DCR_NUM_DATABLKS (34) DHANC 40
DCR_NUM_SYMBOLS (38) DHANC 40
DCR_PREFIX (0) DHANC 40
DCR_PREV (14) DHANC 40
DCR_SYMBOL_BLOCK_MGR (4C) DHANC 40
DCR_SYMBOL_MANAGER (44) DHANC 40
DCR_SYMBOL_SIZE (40) DHANC 40
DCR_SYMBOL_STORAGE_MGR (48) DHANC 40
DCR_SYMBOL_TABLE (44) DHANC 40
DD_BROWSEVAL_SP 8 DDCBC 37
DD_CATALOG_TYPE 8 DDCBC 38
DD_GENERAL_SP 8 DDCBC 37
DD_GLOBAL_LOCK 8 DDCBC 37
DD_LOCK_PREFIX 4 DDCBC 37
DD_SUBPOOL_PREFIX 4 DDCBC 37
DDA (0) DDCBC 36

DDA_ARROW (2) DDCBC 36
DDA_BLOCK_NAME (8) DDCBC 36
DDA_BROWSE_SUBPOOL (20) DDCBC 36
DDA_CICS_BITS 36
DDA_COLD_START 36
DDA_DFH (3) DDCBC 36
DDA_DIRECTORY_LIST (10) DDCBC 36
DDA_DOMID (6) DDCBC 36
DDA_END 36
DDA_GENERAL_SUBPOOL (18) DDCBC 36
DDA_GLOBAL_LOCK (28) DDCBC 36
DDA_IDIRECTORYCLASS (10) DDCBC 36
DDA_LENGTH (0) DDCBC 36
DDA_PREFIX (0) DDCBC 36
DDA_STATE (14) DDCBC 36
DDB (0) DHANC 41
DDB_ARROW (2) DHANC 41
DDB_BIN_BLOCK (BIT) DHANC 41
DDB_BLOCK_NAME (8) DHANC 41
DDB_CODEPAGE (1C) DHANC 41
DDB_DATA (28) DHANC 41
DDB_DATA_LENGTH (24) DHANC 41
DDB_DFH (3) DHANC 41
DDB_DOMID (6) DHANC 41
DDB_LENGTH (0) DHANC 41
DDB_NEXT_CELEM (10) DHANC 41
DDB_NONBIN_BLOCK 41
DDB_PREFIX (0) DHANC 41
DDB_PREV_CELEM (14) DHANC 41
DDBSC 35
DDCBC 36
DEAD_DS_TCBS (14) DSANC 54
DEADLOCK_DELAYED 1 DSTSK 67
DEADLOCK_IMMEDIATE 1 DSTSK 67
DEALLOCATE_TYPE (24) CPCPS 32
DECHAINED (1D) L2SR 251
declarations
    handle manager declarations, PGHM 293
    parameter area declarations, DUFP 76
DEFAULT_APPLID_NAME 8 PAA 284
DEFAULT_BUFFERS 4 TSA 381
DEFAULT_CATALOG_MODULE 1 LDCBS 176
DEFAULT_DSA_LIMIT 4 SMDCC 362
DEFAULT_DSA_RPS_TARGET 4 LDCBS 176
DEFAULT_EDSA_LIMIT 4 SMDCC 362
DEFAULT_EDSA_RPS_TARGET 4 LDCBS 176
DEFAULT_EXECUTION_KEY 1 LDCBS 176
DEFAULT_LANG_PTR (24) MEPS 257
DEFAULT_LANGUAGE (18) MEPS 257
DEFAULT_LANGUAGE_CODE (19) MEPS 257
DEFAULT_PAGESIZE 2 STUCB 378
DEFAULT_PROGRAM_ATTRIBUTE 1 LDCBS 176
DEFAULT_PROGRAM_TYPE 1 LDCBS 175
DEFAULT_PROGRAM_USAGE 1 LDCBS 176
DEFAULT_REQUIRED_AMODE 1 LDCBS 176
DEFAULT_REQUIRED_RMODE 1 LDCBS 176
DEFAULT_STORAGE_FACTOR 1 LDCBS 176
DEFAULT_STRINGS 4 TSA 381
DEFAULT_SUSPRES_AREA (0) DSTSK 64
DEFER (1C) L2SL 241
DEFER_FORCE_FLAG (240) L2BS 217
DEFER_FORCE_FLAG (240) L2SR 249
DEFER_FORCE_INTERVAL 250
DEFERRAL_ACTIVE 1 L2SR 251
DEFERRAL_OVER 1 L2SR 251
definition
    message table definition, MEMMS 252
    resource definition anchor block, RDAB 299
    resource definition recovery definitions, RRAB 341
    resource definition update block, RDUB 300
    transaction manager transaction definition, XMXDC 441
definitions
    resource definition recovery definitions, RRAB 341
DEG_DCL_ID 2 DEGPC 38
DEG_DCL_VERSION 1 DEGPC 38
DEG_PROCESSING_REQS (14) DEGPC 38
DEG_QUEUE_HIWATER (10) DEGPC 38
DEG_QUEUED_REQS (C) DEGPC 38
DEG_REQS_HIWATER (18) DEGPC 38
DEG_STATS_ID (2) DEGPC 38
DEG_STATS_LENGTH (0) DEGPC 38
DEG_STATS_VERSION (4) DEGPC 38
DEG_THREADS 38
DEG_TOTAL_REQS_DEQ (20) DEGPC 38
DEG_TOTAL_REQS_P_TIME (28) DEGPC 38
DEG_TOTAL_REQS_PROCESS (24) DEGPC 38
DEG_TOTAL_REQS_Q_TIME (30) DEGPC 38
DEG_TOTAL_REQS_RCVD (1C) DEGPC 38
DEGPC 38
DELAY_ACTIVE (BIT) DSTSK 66
DELAY_EXPIRED_TIME (98) DSTSK 66
DELAY_OVER_WAIT (BIT) DSTSK 66
DELAY_QUEUE 56
DELAY_QUEUE_HEAD (118) DSANC 56
DELAY_QUEUE_TIME (11C) DSANC 56
DELETE_REQUEST 1 BAACT 20
DELETE_SECONDARY (BIT) L2SL 241
DELETE_TCB_ISSUED (BIT) DSANC 59
DELETE_TCB_RECEIVED (BIT) DSANC 59
DELIVER_DATA (1C) RMLI 304
DELIVER_DATA (84) RMUW 338
DELIVER_DATA (8C4) RMLK 307
DELTA_ROUND 4 TIA 380
description
    dispatcher domain task description, DSTSK 64
descriptor
    connection descriptor, FEP05 117
    document handler template descriptor, DHTL 43
    node descriptor, FEP10 132
    pool descriptor, FEP11 134
    target descriptor, FEP20 147
DESTROY (BIT) L2CH 221
DETACH 1 DSTSK 68
device
    device support extension, FEP08 127
DFH (183) DSANC 57
DFH (3) CPSPS 34
DFH (3) DSANC 54, 58, 60, 61, 62
DFH (3) PRS 296
DFH (3) PTE 298
DFH 3 DDCBC 37
DFH 3 TSA 381
DFH 3 WBANC 413
DFHAPIQ_ARG (0) APIQ 2
DFHCPARH_ADDR (18) CPSPS 34
DFHCPCPS (0) CPCPS 32
DFHCPIR_ADDR (20) CPSPS 34
DFHCPSRH_ADDR (1C) CPSPS 34
DFHD2CSB (0) D2CSB 78
DFHD2CSB_EYECATCHER 14 D2CSB 81
DFHD2ENT (0) D2ENT 81
DFHD2ENT_EYECATCHER 14 D2ENT 85
DFHD2GLB (0) D2GLB 85
DFHD2GLB_COMD_EYECATCHER 14 D2GLB 91
DFHD2GLB_COMD_NAME 8 D2GLB 91
DFHD2GLB_EYECATCHER 14 D2GLB 91
DFHD2GLB_POOL_EYECATCHER 14 D2GLB 91
DFHD2GLB_POOL_NAME 8 D2GLB 91
DFHD2GWA (0) D2GWA 92
DFHD2GWA_EYECATCHER 6 D2GWA 92
DFHD2IDT (0) D2CSB 80
DFHD2LOT (0) D2LOT 93
DFHD2LOT_EYECATCHER 14 D2LOT 95
DFHD2RCT (0) D2ENT 83
DFHD2SS (0) D2SS 96
DFHD2TR (0) D2CSB 80
DFHD2TRN (0) D2TRN 98
DFHD2TRN_EYECATCHER 14 D2TRN 98
DFHDEGPS (0) DEGPC 38
DFHDHPDC (0) DHTL 44
DFHDHTLC (0) DHTL 43
DFHFCPE (0) FCPEC 101
DFHFCPW (0) FCPWC 103
DFHFCQRE (0) FCQRE 104
DFHFCQSE (0) FCQSE 105
DFHFCUP (0) FCUPC 107
DFHFLLB (0) FLLBC 150
DFHKCB (0) KCB 151
DFHLIFO_PLIST (0) KEMHD 162
DFHMEBME_ADDR (904) STUCB 376
DFHMNCR (0) MNCBS 272
DFHPAA (0) PAA 283
DFHPAA_CR (0) PAA 284
DFHRABN (0) RRAB 342
DFHRDAB (0) RDAB 299
DFHRDAL (0) RDAB 299

DFHRDUB (0) RDUB 300
DFHRRAB (0) RRAB 341
DFHSTWRK_ERROR_FLAG (BIT) STUCB 377
DFHSZAI_ARG (0) FEP02 113
DFHSZDAC (0) FEP03 115
DFHSZDAC_LEN 4 FEP03 115
DFHSZDBI (0) FEP04 116
DFHSZDBI_LEN 4 FEP04 116
DFHSZDCD (0) FEP05 117
DFHSZDCD_LEN 4 FEP05 120
DFHSZDCM (0) FEP06 120
DFHSZDCM_LEN 4 FEP06 125
DFHSZDCV (0) FEP07 125
DFHSZDCV_LEN 4 FEP07 126
DFHSZDDS (0) FEP08 127
DFHSZDDS_LEN 4 FEP08 130
DFHSZDEC (0) FEP09 131
DFHSZDEC_LEN 4 FEP09 131
DFHSZDND (0) FEP10 132
DFHSZDND_LEN 4 FEP10 133
DFHSZDPD (0) FEP11 134
DFHSZDPD_LEN 4 FEP11 135
DFHSZDPP (0) FEP12 135
DFHSZDPP_LEN 4 FEP12 136
DFHSZDPS (0) FEP13 136
DFHSZDPS_LEN 4 FEP13 137
DFHSZDQE (0) FEP14 138
DFHSZDQE_LEN 4 FEP14 139
DFHSZDRA (0) FEP15 139
DFHSZDRA_LEN 4 FEP15 140
DFHSZDRB (0) FEP16 140
DFHSZDRB_LEN 4 FEP16 141
DFHSZDRP (0) FEP17 141
DFHSZDRP_LEN 4 FEP17 144
DFHSZDSC (0) FEP18 145
DFHSZDSC_LEN 4 FEP18 145
DFHSZDSR (0) FEP19 146
DFHSZDSR_LEN 4 FEP19 147
DFHSZDTD (0) FEP20 147
DFHSZDTD_LEN 4 FEP20 148
DFHSZSPS (0) FEP21 148
DFHTIA (0) TIA 378
DFHUSGPS (0) USGPS 409
DFHWBEPC (0) WBEPC 421
DFHXSSS (0) XSSS 451
DH_ARROW (2) DDCBC 36
DH_BLOCK_NAME (8) DDCBC 36
DH_BROWSETREE (44) DDCBC 36
DH_CICS_BITS (10) DDCBC 36
DH_CURRENT_BROWSES (40) DDCBC 36
DH_DBB_SP 8 DHANC 42
DH_DCR_SP 8 DHANC 42
DH_DDB_SP 8 DHANC 42
DH_DELETES (3C) DDCBC 36
DH_DFH (3) DDCBC 36
DH_DIRKEYLENGTH (28) DDCBC 36
DH_DIRNAME (24) DDCBC 36
DH_DOA_SP 8 DHANC 42
DH_DOMID (6) DDCBC 36
DH_END (48) DDCBC 36
DH_HASHELEMS (30) DDCBC 36
DH_HASHSIZE (2C) DDCBC 36
DH_HASHTABLE (34) DDCBC 36
DH_IBROWSESEQ 36
DH_IDIRECTORY (24) DDCBC 36
DH_ILOOKUPMAP (2C) DDCBC 36
DH_LENGTH (0) DDCBC 36
DH_LOCAL_LOCK (18) DDCBC 36
DH_LOCK_NAME 8 DHANC 42
DH_NEXT (10) DDCBC 36
DH_PREFIX (0) DDCBC 36
DH_PREV (14) DDCBC 36
DH_REHASH 36
DH_STATE_INITIALISED 1 DHANC 42
DH_STATE_INITIALISING 1 DHANC 42
DH_STATE_QUIESCED 1 DHANC 42
DH_STATE_QUIESCING 1 DHANC 42
DH_STATE_TERMINATED 1 DHANC 42
DH_STB_LENGTH 4 DHANC 42
DH_STB_SP 8 DHANC 42
DH_SUBPOOL (1C) DDCBC 36
DHA (0) DHANC 39
DHA_COLD_START (BIT) DHANC 39
DHA_DBB_SPTOKEN (30) DHANC 39
DHA_DCB_SPTOKEN (38) DHANC 39
DHA_DCR_SPTOKEN (40) DHANC 39
DHA_DDB_SPTOKEN (48) DHANC 39
DHA_DEFAULT_CODEPAGE (18) DHANC 39
DHA_DEFAULT_CODEPAGE_LEN 39
DHA_DH_STATE (10) DHANC 39
DHA_DOA_SPTOKEN (50) DHANC 39
DHA_END (80) DHANC 40
DHA_EYE_CATCHER 14 DHANC 42
DHA_FIRST_DOA (78) DHANC 40
DHA_FLAGS (11) DHANC 39
DHA_GENERAL_SPTOKEN (28) DHANC 39
DHA_LAST_DOA (7C) DHANC 40
DHA_LENGTH (0) DHANC 39
DHA_LOCK_TOKEN (20) DHANC 39
DHA_NUM_DOCUMENTS (14) DHANC 39
DHA_PDS_DCB_FIRST (70) DHANC 40
DHA_PDS_DCB_LAST (74) DHANC 40
DHA_PREFIX (0) DHANC 39
DHA_PREFIX_TEXT (2) DHANC 39
DHA_STB_SPTOKEN (58) DHANC 39
DHA_TEMPLATE_DCB_CHAIN (70) DHANC 40
DHA_TLD_DHT1_DIRTOKEN (68) DHANC 39
DHA_TLD_DHT2_DIRTOKEN (6C) DHANC 39
DHA_TLD_LOCK_TOKEN (24) DHANC 39
DHA_TLD_SPTOKEN (60) DHANC 39
DHA_XRSINDI_ACTIVE (BIT) DHANC 39
DHANC 39
DHPD_ABEND_EXIT_PTR (3C) DHTL 44
DHPD_ABEND_EXIT_RTN (5A) DHTL 45
DHPD_AMODE24_EXIT_ROUTINES (48) DHTL 45
DHPD_ARROW (2) DHTL 44
DHPD_BLOCK_NAME (8) DHTL 44
DHPD_DCB_DESCRIPTOR_END (1D8) DHTL 45
DHPD_DCB_NEXT (10) DHTL 44
DHPD_DCB_OPENLIST 44
DHPD_DCB_PREV (14) DHTL 44
DHPD_DDNAME (18) DHTL 44
DHPD_DECB (60) DHTL 45
DHPD_DFH (3) DHTL 44
DHPD_DIRECTORY_DCB (D0) DHTL 45
DHPD_DIRECTORY_DCB_PTR 44
DHPD_DIRECTORY_EOD_RTN (54) DHTL 45
DHPD_DIRECTORY_EODAD_PTR (38) DHTL 44
DHPD_DOMID (6) DHTL 44
DHPD_DSNAME (128) DHTL 45
DHPD_EXIT_LIST (40) DHTL 44
DHPD_EXLST_ABEND_EXIT_CODE (40) DHTL 44
DHPD_EXLST_ABEND_EXIT_PTR (41) DHTL 44
DHPD_EXLST_JFCB_CODE (44) DHTL 44
DHPD_EXLST_JFCB_PTR (45) DHTL 45
DHPD_FILETYPE 44
DHPD_FILETYPE_HFS 4 DHTL 45
DHPD_FILETYPE_PDS 4 DHTL 45
DHPD_FILETYPE_PDSE 4 DHTL 45
DHPD_FLAG1 (25) DHTL 44
DHPD_IO_ERROR_RTN (48) DHTL 45
DHPD_JFCB (128) DHTL 45
DHPD_LENGTH (0) DHTL 44
DHPD_MEMBER_DCB (78) DHTL 45
DHPD_MEMBER_DCB_PTR 44
DHPD_MEMBER_EOD_RTN (4E) DHTL 45
DHPD_MEMBER_EODAD_PTR (34) DHTL 44
DHPD_PREFIX (0) DHTL 44
DHPD_STATUS (20) DHTL 44
DHPD_SYNAD_PTR (30) DHTL 44
DHTL 43
DHTL_APPEND_CRLF (BIT) DHTL 43
DHTL_ARROW (2) DHTL 43
DHTL_BLDL_DATA (50) DHTL 43
DHTL_BLOCK_NAME (8) DHTL 43
DHTL_CONCATENATION_NO (5B) DHTL 43
DHTL_DDNAME (72) DHTL 43
DHTL_DFH (3) DHTL 43
DHTL_DOCTEMPLATE (10) DHTL 43
DHTL_DOMID (6) DHTL 43
DHTL_EXITPGM_DESCRIPTOR (50) DHTL 44
DHTL_FILE_DESCRIPTOR (50) DHTL 43
DHTL_LENGTH (0) DHTL 43
DHTL_LIBRARY_TYPE (5C) DHTL 43
DHTL_MEMBER_CURRENT_SIZE (6C) DHTL 43
DHTL_MEMBER_DATA (5E) DHTL 43

DHTL_MEMBER_DATE1 43
DHTL_MEMBER_DATE2 (66) DHTL 43
DHTL_MEMBER_HHMM (6A) DHTL 43
DHTL_MEMBER_INITIAL_SIZE (6E) DHTL 43
DHTL_MEMBER_LEN (5D) DHTL 43
DHTL_MEMBER_MODLEVEL (5F) DHTL 43
DHTL_MEMBER_MODLN (70) DHTL 43
DHTL_MEMBER_NAME (50) DHTL 43
DHTL_MEMBER_TTR (58) DHTL 43
DHTL_MEMBER_USERID (72) DHTL 43
DHTL_MEMBER_VERSION (5E) DHTL 43
DHTL_PDS_DCB_DESCRIPTOR 43
DHTL_PDS_DESCRIPTOR (50) DHTL 43
DHTL_PREFIX (0) DHTL 43
DHTL_PROGRAM_DESCRIPTOR (50) DHTL 44
DHTL_RESOURCE_NAME (50) DHTL 43
DHTL_TDQUEUE_DESCRIPTOR (50) DHTL 44
DHTL_TEMPLATE_BODY 43
DHTL_TEMPLATE_END (80) DHTL 44
DHTL_TEMPLATE_EXITPGM (50) DHTL 44
DHTL_TEMPLATE_FILENAME (50) DHTL 44
DHTL_TEMPLATE_FLAGS (4A) DHTL 43
DHTL_TEMPLATE_NAME (18) DHTL 43
DHTL_TEMPLATE_PGMNAME (50) DHTL 44
DHTL_TEMPLATE_TDQNAME (50) DHTL 44
DHTL_TEMPLATE_TSQNAME (50) DHTL 44
DHTL_TEMPLATE_TYPE (48) DHTL 43
DHTL_TSQUEUE_DESCRIPTOR (50) DHTL 44
DHTL_TYPE_BINARY (BIT) DHTL 43
DHTL_TYPE_EBCDIC (BIT) DHTL 43
DIMENSION (24) SOA 371
directory
  directory manager building blocks, DDBSC 35
  directory manager structures, DDCBC 36
DIRHEAD (0) DDCBC 36
DISPATCH_PRIORITY 65
DISPATCH_PRIORITY_BIN 65
DISPATCH_REQUEST 1 BAACT 20
DISPATCHABLE 1 DSTSK 68
DISPATCHABLE_CHAIN (1C) DSANC 58
dispatcher
  dispatcher domain anchor block, DSANC 53
  dispatcher domain task description, DSTSK 64
DISPATCHER_STATE (10) DSANC 54
dm
  dm authorised facility state, DMAFC 45
DMAF_DELETE 1 DMAFC 46
DMAF_DELETE_ENF_ERROR 1 DMAFC 46
DMAF_DISASTER 1 DMAFC 46
DMAF_DUPLICATE_REQUEST 1 DMAFC 46
DMAF_ENF_ANCHOR (C) DMAFC 46
DMAF_ENF_REASON (8) DMAFC 46
DMAF_EXCEPTION 1 DMAFC 46
DMAF_FESTAE_FAIL 1 DMAFC 46
DMAF_FUNCTION 46
DMAF_GETMAIN_D_FAIL 1 DMAFC 46
DMAF_GETMAIN_S_FAIL 1 DMAFC 46
DMAF_INVALID 1 DMAFC 46
DMAF_INVALID_FUNCTION 1 DMAFC 46
DMAF_LISTEN 1 DMAFC 46
DMAF_LISTEN_ENF_ERROR 1 DMAFC 46
DMAF_LISTEN_INACTIVE 1 DMAFC 46
DMAF_NOT_AUTHED 1 DMAFC 46
DMAF_OK 1 DMAFC 46
DMAF_PLIST (0) DMAFC 46
DMAF_PLISTLEN (0) DMAFC 46
DMAF_REASON (7) DMAFC 46
DMAF_RESPONSE 46
DMAF_STATE (0) DMAFC 45
DMAF_SVC_CALL_A_FAIL 1 DMAFC 46
DMAF_SVC_CALL_D_FAIL 1 DMAFC 46
DMAF_SVC_RESPONSE (10) DMAFC 46
DMAFC 45
DMAFS_ASCB (18) DMAFC 45
DMAFS_ENF_ANCHOR (10) DMAFC 45
DMAFS_ENF_DTOKEN (1C) DMAFC 45
DMAFS_EYE (2) DMAFC 45
DMAFS_LEN (0) DMAFC 45
DMAFS_TCB (14) DMAFC 45
DMCB1 47
DMCB2 49
DMCB3 50
DMCB4 51
DMENC 52
DMPH_APPLICATIONS_FINISHED 2 DMCB1 49
DMPH_BASIC_FUNCTIONS_AVAILABLE 2 DMCB1 48
DMPH_BOTTOM 2 DMCB1 49
DMPH_CSA_AVAILABLE 2 DMCB1 48
DMPH_CWA_AVAILABLE 2 DMCB1 48
DMPH_DEFAULT_USER_AVAILABLE 2 DMCB1 48
DMPH_ESM_AVAILABLE 2 DMCB1 48
DMPH_GLOBAL_CATALOG_AVAILABLE 2 DMCB1 48
DMPH_GLOBAL_CATALOG_FOR_RM 2 DMCB1 48
DMPH_LANGUAGE_ENVIRONMENT_READY 2 DMCB1 48
DMPH_PRE_INIT_COMPLETE 2 DMCB1 48
DMPH_PRIMARY_TERMINATED 2 DMCB1 48
DMPH_RECOVERY_ACTIVE 2 DMCB1 48
DMPH_RM_CLIENTS_REGISTERED 2 DMCB1 48
DMPH_RM_STARTUP_TYPE_KNOWN 2 DMCB1 48
DMPH_SHUTDOWN_STATS_READY 2 DMCB1 49
DMPH_STATISTICS_AVAILABLE 2 DMCB1 48
DMPH_STATISTICS_UNAVAILABLE 2 DMCB1 49
DMPH_SYSTEM_FUNCTIONS_AVAILABLE 2 DMCB1 48
DMPH_SYSTEM_LOG_AVAILABLE 2 DMCB1 48
DMPH_TIMER_AVAILABLE 2 DMCB1 48
DMPH_TOP 2 DMCB1 48
DMPH_TS_BASIC_RECOVERY_COMPLETE 2 DMCB1 48
DMPH_XM_ATTACH_AVAILABLE 2 DMCB1 48
DOA (0) DHANC 40
DOA_ARROW (2) DHANC 40
DOA_BLOCK_NAME (8) DHANC 40
DOA_DFH (3) DHANC 40
DOA_DOMID (6) DHANC 40
DOA_FIRST_DCR (18) DHANC 40
DOA_LAST_DCR (1C) DHANC 40
DOA_LENGTH (0) DHANC 40
DOA_NEXT (10) DHANC 40
DOA_PREFIX (0) DHANC 40
DOA_PREV (14) DHANC 40
DOA_TRANNUM (20) DHANC 40
DOA_TRANSID (24) DHANC 40
document
  document handler anchor block, DHANC 39
  document handler template descriptor, DHTL 43
DOH_ARROW (2) KECB 155
DOH_BLOCK_NAME (8) KECB 155
DOH_DFH (3) KECB 155
DOH_DOMID (6) KECB 155
DOH_END (20) KECB 155
DOH_ENTRY_LENGTH (18) KECB 155
DOH_LENGTH (0) KECB 155
DOH_PREFIX (0) KECB 155
DOH_TABLE_END (14) KECB 155
DOH_TABLE_START (10) KECB 155
DOM_AFFINITY 155
DOM_AFFINITY_CO (BIT) KECB 155
DOM_AFFINITY_FO (BIT) KECB 155
DOM_AFFINITY_QR (BIT) KECB 155
DOM_AFFINITY_RO (BIT) KECB 155
DOM_AFFINITY_STEP (BIT) KECB 155
DOM_ANCHOR 155
DOM_DEFAULT_RECOVERY (1C) KECB 155
DOM_GATE_ENTRY (28) KECB 155
DOM_GATE_TABLE (28) KECB 155
DOM_GATE_TABLE_NAME (20) KECB 155
DOM_INDEX (8) KECB 155
DOM_NAME (0) KECB 155
DOM_SPECIAL_TRACE (18) KECB 155
DOM_STANDARD_TRACE (14) KECB 155
DOM_STATE (C) KECB 155
DOM_STATE_FLAG (C) KECB 155
DOM_TERMINATED (BIT) KECB 155
domain
  dce services domain global statistics, DEGPC 38
  dispatcher domain anchor block, DSANC 53
  dispatcher domain task description, DSTSK 64
  domain manager anchor block, DMCB1 47
  domain manager browse cursor, DMCB2 49
  domain manager enf state, DMENC 52
  domain manager wait queue element, DMCB3 50
  domain record, DMCB4 51
  enqueue domain anchor block, NQA 275
  enqueue domain browse element, NQB 276
  enqueue domain browse owner extension, NQOX 279
  enqueue domain browse waiter extension, NQWX 282

domain *(continued)*
  enqueue domain enqueue pool, NQPL 280
  enqueue domain queue element area, NQEA 277
  loader domain control blocks, LDCBS 164
  lock manager domain anchor block, LMCB1 204
  lock manager domain quickcell headers, LMCB2 206
  logger domain anchor block, LGANC 188
  message domain anchor block, MEPS 257
  monitoring domain control blocks, MNCBS 262
  parameter manager domain anchor block, PAA 283
  partner domain static storage area, PRS 296
  recovery manager domain management instance, RMDM 301
  security domain anchor block, XSANC 448
  security domain transaction data, XSXD 455
  security domain transaction token, XSXT 456
  statistics domain anchor block, STCB1 374
  timer domain anchor block, TIA 378
  transaction manager domain anchor block, XMANC 435
  user domain anchor block, USANC 405
  user domain statistics, USGPS 409
  user domain transaction data, USXD 410
  user domain transaction token, USXT 411
  user domain user data block, UDB 403
  web domain anchor block, WBANC 412
DOMAIN (18) SOA 371, 372
DOMAIN (1C) SOA 372
DOMAIN_ENTRY (0) KECB 155
DOMAIN_HEADER (0) KECB 155
DOMAIN_OWNER 66
DOMAIN_RECORD (0) DMCB4 51
DOMAIN_STATUS (60) STCB1 374
DOMID (186) DSANC 57
DOMID (6) CPSPS 34
DOMID (6) DSANC 54, 58, 60, 61, 62
DOMID (6) PRS 296
DOMID (6) PTE 298
DOUBLE_CHAIN (0) DSANC 60
DPL_EXEC_PLIST (178) XCCBC 433
DR_ARROW (2) DMCB4 51
DR_BLOCK_NAME (8) DMCB4 51
DR_DFH (3) DMCB4 51
DR_DOMAIN_ID (1C) DMCB4 51
DR_DOMAIN_TOKEN (10) DMCB4 51
DR_DOMID (6) DMCB4 51
DR_LENGTH (0) DMCB4 51
DR_PREFIX (0) DMCB4 51
DR_PROG_NAME (14) DMCB4 51
DS_CELL_PAM (0) DSANC 61
DS_EXTENSION_PAM (0) DSANC 62
DS_FLAGS 54
DS_SUSPEND_PAM (0) DSANC 62
DS_TASK_PAM (0) DSANC 62
DS_TCB (0) DSANC 58
DS_TCB_FLAGS (4D) DSANC 59
DS_TCB_FLAGS2 (4E) DSANC 59
DS_TCB_PART1 (0) DSANC 58
DS_TCB_PART2 (18) DSANC 58
DS_TCB_PART3 (28) DSANC 58
DSA (2C4) LDCBS 171
DSA (68) LDCBS 165
DSA_EXTENT_SHIFT 4 SMDCC 362
DSA_MULTIPLE 4 SMDCC 362
DSA2 (48) LDCBS 170
 DSANC 53
DSAUSB (0) DSANC 61
DSAUSB_END (A4) DSANC 61
DSAUTB (0) DSANC 61
DSAUTB_END (68) DSANC 61
DSCSA_WORK (158) DSANC 56
DSIT_LOCK_TOKEN (600) DSANC 57
DSPSWAP (98) DSANC 61
DSPXADD (30) DSANC 61
DSPXENAB (9C) DSANC 61
DSPXENT (10) DSANC 61
DSSEYECATCH (0) DSANC 61
DSSREGSAV (50) DSANC 61
DST_DS_TCB_ADDR (10) DSANC 61
 DSTBA 63
DSTEYECATCH (0) DSANC 61
DSTPEXAD (5C) DSANC 61
DSTREGSAV (14) DSANC 61
 DSTSK 64
DSTUSER_PARM (60) DSANC 61
DTA_XM_TXN (80) DSTSK 66
DTB (0) DHANC 41
DTB_NEXT_TEMPLATE (0) DHANC 41
DTB_PREV_TEMPLATE (4) DHANC 41
DTB_TEMPLATE_DATA (8) DHANC 41
DTCHD_ARROW (2) DTCPS 68
DTCHD_BLOCK (0) DTCPS 68
DTCHD_CALLER_RB (1C) DTCPS 68
DTCHD_DFHDT (3) DTCPS 68
DTCHD_ID (8) DTCPS 68
DTCHD_LEN (0) DTCPS 68
DTCHD_LX_MAP (20) DTCPS 68
DTCHD_PREFIX (0) DTCPS 68
DTCHD_VECTOR_DESC (10) DTCPS 68
DTCHD_VECTOR_HI_ACTIVE_INDEX (18) DTCPS 68
DTCHD_VECTOR_PTR (10) DTCPS 68
DTCHD_VECTOR_SIZE (14) DTCPS 68
DTCON_APPLID (10) DTCPS 69
DTCON_ASID (4) DTCPS 69
DTCON_COUNT (0) DTCPS 69
DTCON_FILE_NAME (18) DTCPS 69
DTCON_FILE_REUSE (8) DTCPS 69
DTCON_FILE_TOKEN (C) DTCPS 69
DTCON_INFO (6) DTCPS 69
DTCON_LX (6) DTCPS 69
DTCON_VECTOR (0) DTCPS 69
 DTCPS 68
DTDUM_ARROW (2) DTLPS 69
DTDUM_BLOCK (0) DTLPS 69
DTDUM_CHAIN (18) DTLPS 69
DTDUM_CHANGES (1C) DTLPS 70
DTDUM_DFHDT (3) DTLPS 69
DTDUM_HEADER_PTR (20) DTLPS 70
DTDUM_ID (8) DTLPS 69
DTDUM_LEN (0) DTLPS 69
DTDUM_NAME (10) DTLPS 69
DTDUM_NEXT (18) DTLPS 69
DTDUM_PREFIX (0) DTLPS 69
DTFIL_A_FLAGS 71
DTFIL_ARROW (2) DTLPS 71
DTFIL_ATTRS (2C) DTLPS 71
DTFIL_ATTRS_LEN (28) DTLPS 71
DTFIL_AVAILABLE (BIT) DTLPS 71
DTFIL_BLOCK (0) DTLPS 71
DTFIL_CHAIN (18) DTLPS 71
DTFIL_CONTINUE (BIT) DTLPS 71
DTFIL_DFHDT (3) DTLPS 71
DTFIL_ENABLED (BIT) DTLPS 71
DTFIL_FLAGS (24) DTLPS 71
DTFIL_ID (8) DTLPS 71
DTFIL_INITIATOR (BIT) DTLPS 71
DTFIL_LEN (0) DTLPS 71
DTFIL_NAME (10) DTLPS 71
DTFIL_NEXT (18) DTLPS 71
DTFIL_PREFIX (0) DTLPS 71
DTFIL_REUSE_COUNT (1C) DTLPS 71
DTFIL_TABLE_PTR (20) DTLPS 71
DTHDR_ARROW (2) DTLPS 69
DTHDR_BACKOUT_POOL (40) DTLPS 69
DTHDR_BLOCK (0) DTLPS 69
DTHDR_DATA_SPACE_PTR (48) DTLPS 69
DTHDR_DFHDT (3) DTLPS 69
DTHDR_DTFOR_EP (10) DTLPS 69
DTHDR_FILE_COUNT (34) DTLPS 69
DTHDR_FILE_FREE (30) DTLPS 69
DTHDR_FILE_HEAD (28) DTLPS 69
DTHDR_FILE_INFO (28) DTLPS 69
DTHDR_FILE_POOL (2C) DTLPS 69
DTHDR_ID (8) DTLPS 69
DTHDR_LEN (0) DTLPS 69
DTHDR_LOAD_ID (3C) DTLPS 69
DTHDR_MAX_ATTRS_LEN (38) DTLPS 69
DTHDR_PREFIX (0) DTLPS 69
DTHDR_PRIMARY_ALET (44) DTLPS 69
DTHDR_RE_WORK 69
DTHDR_RECMAN_EP (14) DTLPS 69
DTHDR_TABLE_COUNT (24) DTLPS 69
DTHDR_TABLE_FREE (20) DTLPS 69
DTHDR_TABLE_HEAD (18) DTLPS 69
DTHDR_TABLE_INFO (18) DTLPS 69
DTHDR_TABLE_POOL (1C) DTLPS 69
DTIMOUT (B0) DSTSK 66
 DTLPS 69

DTN 393
DTN_DOWN (18) TSNM 393
DTN_DOWN_COUNT (17) TSNM 393
DTN_END (58) TSNM 393
DTN_NAME (0) TSNM 393
DTN_OFFSET (14) TSNM 393
DTN_SHIFT (16) TSNM 393
DTN_SUBTRACT (15) TSNM 393
DTN_UP (10) TSNM 393
DTRGN_ALET_LIST_PTR (38) DTSPS 73
DTRGN_ANCHOR (0) DTSPS 72
DTRGN_ARROW (2) DTSPS 72
DTRGN_CONNECT_INFO (14) DTSPS 72
DTRGN_CONNECT_PTR (1C) DTSPS 73
DTRGN_DFHDT (3) DTSPS 72
DTRGN_DTAM_LENGTH (44) DTSPS 73
DTRGN_DTAM_ORIGIN (48) DTSPS 73
DTRGN_EOM_RESMGR_DELETE_ACTIVE (BIT) DTSPS 73
DTRGN_EOM_TOKEN (2C) DTSPS 73
DTRGN_EXIT_WORKA_PTR (3C) DTSPS 73
DTRGN_FLAGS (40) DTSPS 73
DTRGN_HEADER_PTR (20) DTSPS 73
DTRGN_HOME_STOKEN (30) DTSPS 73
DTRGN_ID (8) DTSPS 72
DTRGN_LEN (0) DTSPS 72
DTRGN_LOOKUP_EP (18) DTSPS 72
DTRGN_PREFIX (0) DTSPS 72
DTRGN_RECMAN_EP (24) DTSPS 73
DTRGN_REMOTE_PTR (14) DTSPS 72
DTRGN_SERVER_INFO (20) DTSPS 73
DTRGN_SERVER_PTR (28) DTSPS 73
DTRGN_SYSTEM_PTR (10) DTSPS 72
DTRGN_TRANSWAP (BIT) DTSPS 73
DTRHD_ARROW (2) DTRPS 72
DTRHD_BLOCK (0) DTRPS 72
DTRHD_DFHDT (3) DTRPS 72
DTRHD_DTAOR_EP 72
DTRHD_ID (8) DTRPS 72
DTRHD_LEN (0) DTRPS 72
DTRHD_PREFIX (0) DTRPS 72
DTRPS 72
DTSEC_ARROW (2) DTXPS 74
DTSEC_BLOCK (0) DTXPS 74
DTSEC_DEFAULT_USERID (18) DTXPS 74
DTSEC_DFHDT (3) DTXPS 74
DTSEC_FC_CLASS_NAME (2C) DTXPS 74
DTSEC_FC_CLASS_NAME_LENGTH 74
DTSEC_ID (8) DTXPS 74
DTSEC_LEN (0) DTXPS 74
DTSEC_PREFIX (0) DTXPS 74
DTSEC_RESNAME_PREFIX (20) DTXPS 74
DTSEC_RESNAME_PREFIX_LENGTH (29) DTXPS 74
DTSEC_SERVER_USERID (10) DTXPS 74
DTSPS 72
DTSRV_APPLID (10) DTSPS 73
DTSRV_ARROW (2) DTSPS 73
DTSRV_ASID (20) DTSPS 73
DTSRV_DFHDT (3) DTSPS 73
DTSRV_DTAM_LENGTH (30) DTSPS 73
DTSRV_DTAM_ORIGIN (34) DTSPS 73
DTSRV_ELEMENT (0) DTSPS 73
DTSRV_ET_TOKEN (24) DTSPS 73
DTSRV_ID (8) DTSPS 73
DTSRV_LEN (0) DTSPS 73
DTSRV_LX (22) DTSPS 73
DTSRV_NEXT (18) DTSPS 73
DTSRV_PREFIX (0) DTSPS 73
DTSRV_SEC_EP (28) DTSPS 73
DTSRV_SEC_TOKEN (2C) DTSPS 73
DTSRV_SYSTEM_PTR (1C) DTSPS 73
DTSYS_ACTIVE_CLOCK (10) DTSPS 72
DTSYS_ANCHOR (0) DTSPS 72
DTSYS_ARROW (2) DTSPS 72
DTSYS_CONNECTS_IN_FLIGHT (1C) DTSPS 72
DTSYS_DFHDT (3) DTSPS 72
DTSYS_ID (8) DTSPS 72
DTSYS_LEN (0) DTSPS 72
DTSYS_PREFIX (0) DTSPS 72
DTSYS_SERVER_HEAD (18) DTSPS 72
DTTBL_ADD_GAP (BIT) DTLPS 70
DTTBL_ADD_SAVE (6C) DTLPS 70
DTTBL_ARROW (2) DTLPS 70
DTTBL_AVAILABLE (BIT) DTLPS 70
DTTBL_BLOCK (0) DTLPS 70
DTTBL_CHAIN (18) DTLPS 70
DTTBL_CHANGES (1C) DTLPS 70
DTTBL_CMT (BIT) DTLPS 70
DTTBL_DATA_ALET_PTR (90) DTLPS 70
DTTBL_DATA_COUNT (B0) DTLPS 71
DTTBL_DATA_END (A4) DTLPS 71
DTTBL_DATA_FRAME (98) DTLPS 71
DTTBL_DATA_FREE (AC) DTLPS 71
DTTBL_DATA_HEAD (94) DTLPS 70
DTTBL_DATA_HWM (B4) DTLPS 71
DTTBL_DATA_INFO (8C) DTLPS 70
DTTBL_DATA_NEXT (A0) DTLPS 71
DTTBL_DATA_SIZE (A8) DTLPS 71
DTTBL_DATA_SPACE (8C) DTLPS 70
DTTBL_DATA_START (9C) DTLPS 71
DTTBL_DFHDT (3) DTLPS 70
DTTBL_DSNAME (BC) DTLPS 71
DTTBL_DSNAME_LEN (30) DTLPS 70
DTTBL_DSNAME_PTR (2C) DTLPS 70
DTTBL_ENTRY_ALET_PTR (5C) DTLPS 70
DTTBL_ENTRY_COUNT (64) DTLPS 70
DTTBL_ENTRY_HWM (44) DTLPS 70
DTTBL_ENTRY_INFO (5C) DTLPS 70
DTTBL_ENTRY_LIMIT (68) DTLPS 70
DTTBL_ENTRY_POOL (60) DTLPS 70
DTTBL_FILE_COUNT (28) DTLPS 70
DTTBL_FLAGS (24) DTLPS 70
DTTBL_FULL_COUNT (40) DTLPS 70
DTTBL_HEADER_PTR (20) DTLPS 70
DTTBL_ID (8) DTLPS 70
DTTBL_INCOMPLETE (BIT) DTLPS 70
DTTBL_INDEX_ALET_PTR (7C) DTLPS 70
DTTBL_INDEX_COUNT (84) DTLPS 70
DTTBL_INDEX_HWM (88) DTLPS 70
DTTBL_INDEX_INFO 70
DTTBL_INDEX_POOL (80) DTLPS 70
DTTBL_INDEX_ROOT (78) DTLPS 70
DTTBL_KEY_LEN (48) DTLPS 70
DTTBL_KEY_OFFSET (4C) DTLPS 70
DTTBL_LEN (0) DTLPS 70
DTTBL_LOAD_COUNT (38) DTLPS 70
DTTBL_LOAD_DISC (BIT) DTLPS 70
DTTBL_LOAD_DISC_KEY (58) DTLPS 70
DTTBL_LOAD_EOF (BIT) DTLPS 70
DTTBL_LOAD_GAP (BIT) DTLPS 70
DTTBL_LOAD_HIGH_KEY (54) DTLPS 70
DTTBL_LOAD_ID (34) DTLPS 70
DTTBL_MAX_RECLEN (50) DTLPS 70
DTTBL_NAME (10) DTLPS 70
DTTBL_NEXT (18) DTLPS 70
DTTBL_PREFIX (0) DTLPS 70
DTTBL_RECOVERABLE (BIT) DTLPS 70
DTTBL_REJECT_COUNT (3C) DTLPS 70
DTTBL_RETRY_COUNT (B8) DTLPS 71
DTTBL_STATS (38) DTLPS 70
DTTBL_T_FLAGS 70
DTXPS 74
DUF_ADD_INDEX_ENTRY 4 DUFP 77
DUF_ADD_LIST 4 DUFP 77
DUF_ADD_LIST_REVERSE 4 DUFP 77
DUF_ADDRESS (34) DUFP 76
DUF_AFCB_PTR (4) DUFC 75
DUF_ALLOW_ZERO (BIT) DUFP 76
DUF_ANCHOR_PTR (2C) DUFP 76
DUF_ARROW (2) DUFP 76
DUF_BLCK_NAME (8) DUFP 76
DUF_BLOCK_ADDRESS (20) DUFP 76
DUF_BLOCK_LENGTH (24) DUFP 76
DUF_BLOCK_NAME (3C) DUFP 76
DUF_BLOCK_RESOURCE (44) DUFP 76
DUF_BLOCK_RESOURCE2 (C8) DUFP 76
DUF_BLOCK_RESOURCE2_X 76
DUF_BLOCK_TITLE (58) DUFP 76
DUF_BLOCK_TITLE_LENGTH (54) DUFP 76
DUF_BOUNDARY (19) DUFP 76
DUF_BROWSE_TOKEN (30) DUFP 76
DUF_COM (0) DUFC 75
DUF_COM_PTR (10) DUFP 76
DUF_CREATE_LIST 4 DUFP 77
DUF_CREATE_LIST_REVERSE 4 DUFP 77
DUF_DELETE_LIST 4 DUFP 77
DUF_DFH (3) DUFP 76

DUF_DOMAIN_ANCHOR (0) DUFC 75
DUF_DOMAIN_TABLE (0) DUFC 75
DUF_DOMAIN_TABLE_PTR (10) DUFC 75
DUF_DOMID (6) DUFP 76
DUF_DUFF_PTR (114) DUFP 77
DUF_DUMP_HEADER_STCK (F9) DUFP 77
DUF_DUPLICATE_ADDRESS 4 DUFP 78
DUF_EJECT (BIT) DUFP 76
DUF_END_BROWSE 4 DUFP 78
DUF_ERB_EFREE (68) DUFC 75
DUF_ERB_EHEAD (64) DUFC 75
DUF_ERB_IFREE (60) DUFC 75
DUF_ERB_IHEAD (5C) DUFC 75
DUF_FLAGS (15) DUFP 76
DUF_FLAGS2 77
DUF_FORMAT_BLOCK 4 DUFP 77
DUF_FORMAT_BLOCKS (BIT) DUFP 77
DUF_FORMAT_CHECKING (BIT) DUFP 77
DUF_FORMAT_LEVEL 77
DUF_FORMAT_MAIN_STORAGE 4 DUFP 77
DUF_FORMAT_STCK 4 DUFP 77
DUF_FORMAT_SUMMARY (BIT) DUFP 77
DUF_FORMATTING_ERROR 4 DUFP 78
DUF_FUNCTION (14) DUFP 76
DUF_GET_BLOCK 4 DUFP 77
DUF_INDEX_ENTRY_TEXT (58) DUFP 76
DUF_INDEX_ENTRY_TEXT_LENGTH (54) DUFP 76
DUF_INDEX_ENTRY_TYPE 76
DUF_INDEX_ENTRY_TYPE_BLOCK 4 DUFP 77
DUF_INDEX_ENTRY_TYPE_KEYWORD 4 DUFP 77
DUF_INDEX_ENTRY_TYPE_TEXT 4 DUFP 77
DUF_INVALID_ADDRESS 4 DUFP 78
DUF_INVALID_BROWSE_TOKEN 4 DUFP 78
DUF_INVALID_DATA_LEN 4 DUFP 78
DUF_LENGTH (0) DUFP 76
DUF_LINE (58) DUFP 76
DUF_LINES_LEFT_ON_PAGE (108) DUFP 77
DUF_LIST_TOKEN (30) DUFP 76
DUF_LONG_NAME (3C) DUFP 76
DUF_LONG_NAME_X (BIT) DUFP 76
DUF_MESSAGE_TEXT (58) DUFP 76
DUF_MESSAGE_TEXT_LENGTH (54) DUFP 76
DUF_MESSAGE_TYPE (18) DUFP 76
DUF_MSG_FORMATTING_ERROR 4 DUFP 77
DUF_MSG_INVALID_ADDRESS 4 DUFP 77
DUF_MSG_INVALID_DATA 4 DUFP 78
DUF_MSG_INVALID_DATA_LEN 4 DUFP 77
DUF_MSG_INVALID_EYECATCHER 4 DUFP 77
DUF_MSG_INVALID_POINTER 4 DUFP 77
DUF_MSG_LOOP_DETECTED 4 DUFP 77
DUF_MSG_SAA1_INVALID 4 DUFP 77
DUF_MSG_SAA2_INVALID 4 DUFP 77
DUF_MSG_SAAS_DIFFER 4 DUFP 77
DUF_MSG_SAAS_INVALID 4 DUFP 77
DUF_MSG_TMP_GET_NEXT 4 DUFP 77
DUF_MSG_TMP_START_BROWSE 4 DUFP 77
DUF_MSG_UNREFERENCED_PAGE 4 DUFP 77
DUF_MSG_ZERO_ADDRESS 4 DUFP 77
DUF_MSG_ZERO_POINTER 4 DUFP 77
DUF_NDX_FREEHEAD (58) DUFC 75
DUF_NDX_HEAD (28) DUFC 75
DUF_NOT_FOUND 4 DUFP 78
DUF_OFFSET (38) DUFP 76
DUF_OK 4 DUFP 78
DUF_PARMS (0) DUFP 76
DUF_PF3_PRESSED (BIT) DUFP 77
DUF_PRDMP_PARMLIST_PTR (0) DUFC 75
DUF_PREFIX (0) DUFP 76
DUF_PRINT_LINE 4 DUFP 77
DUF_PRINT_MESSAGE 4 DUFP 77
DUF_QUIT_JOB 4 DUFP 78
DUF_RC (1C) DUFP 76
DUF_READ_INDEX (110) DUFP 77
DUF_READ_LIST 4 DUFP 77
DUF_READ_LIST_REVERSE 4 DUFP 77
DUF_READ_PTR (10C) DUFP 77
DUF_READ_TOKEN 77
DUF_SET_PTR (28) DUFP 76
DUF_SEVERITY_LEVEL (17) DUFP 76
DUF_SEVERITY_LEVEL_E 4 DUFP 78
DUF_SEVERITY_LEVEL_I 4 DUFP 78
DUF_SPACE_AFTER (BIT) DUFP 76
DUF_SPACE_BEFORE (BIT) DUFP 76
DUF_START_READ_LIST 4 DUFP 77
DUF_START_READ_LIST_REVERSE 4 DUFP 77
DUF_TABLE_ENTRY_ADDRESS (34) DUFP 76
DUF_TIME_DATE 77
DUF_TIME_DATE_FORMAT (E0) DUFP 77
DUF_TIME_DATE_STCK (F1) DUFP 77
DUF_TMP_END_BROWSE 4 DUFP 77
DUF_TMP_GET_NEXT 4 DUFP 77
DUF_TMP_GET_NEXT_ERROR 4 DUFP 78
DUF_TMP_START_BROWSE 4 DUFP 77
DUF_TMP_START_BROWSE_ERROR 4 DUFP 78
DUF_TMP_TABLE (16) DUFP 76
DUF_TMP_TABLE_AFCT 4 DUFP 78
DUF_TMP_TABLE_AITM 4 DUFP 78
DUF_TMP_TABLE_DCT 4 DUFP 78
DUF_TMP_TABLE_DSN 4 DUFP 78
DUF_TMP_TABLE_DSNA 4 DUFP 78
DUF_TMP_TABLE_DUMY 4 DUFP 78
DUF_TMP_TABLE_FCT 4 DUFP 78
DUF_TMP_TABLE_PFT 4 DUFP 78
DUF_TMP_TABLE_PRT 4 DUFP 78
DUF_TMP_TABLE_TCNT 4 DUFP 78
DUF_TMP_TABLE_TCTE 4 DUFP 78
DUF_TMP_TABLE_TCTN 4 DUFP 78
DUF_TMP_TABLE_TCTS 4 DUFP 78
DUF_TRFCA_PTR 77
DUF_UPPERCASE_REQ 75
DUF_USER_PARMS (14) DUFP 76
DUFC 75
DUFP 76
DUID_PA_LOOP 8 PAA 284
DUID_PA_RECOVERY 8 PAA 284
DUID_PA_SEVERE_ERROR 8 PAA 284
DUID_TI_BADSTCK 8 TIA 380
DUID_TI_LOOP 8 TIA 380
DUID_TI_RECOV 8 TIA 380
DUMMY (0) DDBSC 35
DUMMY_CDCHAIN (18) LDCBS 169
DUMMY_CDE (0) LDCBS 168
DUMMY_CDE_ANCHOR (164) LDCBS 170
DUMMY_CDE_ARROW (2) LDCBS 169
DUMMY_CDE_BLOCK_ID (8) LDCBS 169
DUMMY_CDE_CHAIN (10) LDCBS 169
DUMMY_CDE_CONTENTS (18) LDCBS 169
DUMMY_CDE_DFH (3) LDCBS 169
DUMMY_CDE_DOMAIN (6) LDCBS 169
DUMMY_CDE_LENGTH (0) LDCBS 169
DUMMY_CDE_NEXT (10) LDCBS 169
DUMMY_CDE_POOL_BDY 2 LDCBS 175
DUMMY_CDE_POOL_NAME 8 LDCBS 174
DUMMY_CDE_PREFIX (0) LDCBS 169
DUMMY_CDE_PREV (14) LDCBS 169
DUMMY_CDENTPT (28) LDCBS 169
DUMMY_CDNAME 169
DUMMY_CDXLMJP (2C) LDCBS 169
DUMMY_LOGSTREAM_TOKEN 4 L2SL 241
DUMMY_PRIMARY (BIT) L2CH 220
DUMMY_SECONDARY_STREAM 4 L2CH 224
DUMMY_XTLMSBAA (C) LDCBS 169
DUMMY_XTLMSBLA 169
DUMMY_XTLST (0) LDCBS 169
dump
  dump formatting communication area, DUFC 75
DWORDUP 4 PAA 284
DXE (0) SMDCC 355
DXE_DSA_NAME 355
DXE_DXGP (18) SMDCC 355
DXE_EXTENT_END (14) SMDCC 355
DXE_EXTENT_START (10) SMDCC 355
DXE_FLAGS (20) SMDCC 355
DXE_IDENTIFIED (BIT) SMDCC 355
DXE_LD_CHECK_NEXT (8) SMDCC 355
DXE_LD_CHECK_PREV (C) SMDCC 355
DXE_NEXT (0) SMDCC 355
DXE_PPXP (1C) SMDCC 355
DXE_PREV (4) SMDCC 355
DXEBLOCK_NAME 8 SMDCC 356
DXEBLOCK_SIZE 4 SMDCC 363
DXG (0) SMDCC 355
DXG_ADDR (8) SMDCC 355
DXG_LEN (C) SMDCC 355
DXG_MVS_KEY (11) SMDCC 355
DXG_MVS_SUBPOOL (10) SMDCC 355

DXG_NEXT (0) SMDCC 355
DXG_PREV (4) SMDCC 355
DXH (0) SMDCC 354
DXH_ABOVE_EXTENT_HEAD (C0) SMDCC 355
DXH_ABOVE_GETMAIN_HEAD 355
DXH_ABOVE_LD_CHECK_HEAD (E8) SMDCC 355
DXH_ALLOCATE_DSA_EXTENT_REQUESTS (120) SMDCC 355
DXH_ARROW (2) SMDCC 354
DXH_BELOW_EXTENT_HEAD (48) SMDCC 354
DXH_BELOW_GETMAIN_HEAD 354
DXH_BELOW_LD_CHECK_HEAD (70) SMDCC 355
DXH_BLOCK_NAME (8) SMDCC 354
DXH_DFH (3) SMDCC 354
DXH_DOMID (6) SMDCC 354
DXH_EXTENT_GETMAINS (124) SMDCC 355
DXH_EXTENT_GETMAINS_EXPLICIT (128) SMDCC 355
DXH_EXTENT_GETMAINS_NOSTG (134) SMDCC 355
DXH_EXTENT_GETMAINS_SINGLE (12C) SMDCC 355
DXH_EXTENT_GETMAINS_VTYPE (130) SMDCC 355
DXH_EXTENT_MULTIPLE_ABOVE (1C) SMDCC 354
DXH_EXTENT_MULTIPLE_BELOW (18) SMDCC 354
DXH_FLAGS 354
DXH_FREE_HEAD (14) SMDCC 354
DXH_GET_DSALIM_REQUESTS (118) SMDCC 355
DXH_GET_DSALIM_REQUESTS_NOSTG (11C) SMDCC 355
DXH_LENGTH (0) SMDCC 354
DXH_LOC_EXPLICIT (BIT) SMDCC 354
DXH_PREFIX (0) SMDCC 354
DXH_REENTRANT_PROGRAM_PROTECT (BIT) SMDCC 354
DXH_STORAGE_PROTECT (BIT) SMDCC 354
DXH_TRACEP 355
DXH_TRANSACTION_ISOLATION (BIT) SMDCC 354
DXH_VGETSP (114) SMDCC 355

E

EBCDIC_VALUE (11) WBANC 413
ECB_CLEAR 4 L2HS 231
ECB_LIST 1 DSTSK 68
ECB_POINTER (38) SOA 371
ECB_POSTED 4 L2HS 231
ECB_Q_DW (98) DSANC 55
ECB_SINGLE 1 DSTSK 68
ECBPARM (70) DSTSK 66
ECBPARM_TYPE (75) DSTSK 66
ECDSA 4 SMDCC 363
ECDSA_NAME 5 LDCBS 175
ECDSA_NAME 8 SMDCC 363
EDSA_EXTENT_SHIFT 4 SMDCC 362
EDSA_MULTIPLE 4 SMDCC 362
ELAPSED 0 STUCB 378
element
  domain manager wait queue element, DMCB3 50
  enqueue domain browse element, NQB 276
  enqueue domain queue element area, NQEA 277
  file control cfdt pool element, FCPEC 101
  file control cfdt pool wait element, FCPWC 102
  file control quiesce receive element, FCQRE 104
  file control quiesce send element, FCQSE 105
  transaction manager resource lock element, XMRLC 440
  transaction manager tran. browse element, XMXBC 441
  work queue element, FEP14 138
ELEN (1C) DDBSC 35
ELPA_NAME 5 LDCBS 175
EMPTY_LOG_STREAM 4 L2HS 231
EMPTY_STREAM 251
EMPTY_STREAM 4 L2BL 211
ENABLESTATUS (0) BAPT 23
END_DELIVERY (20) RMLI 304
END_DELIVERY (88) RMUW 338
END_DELIVERY (8C8) RMLK 307
END_KEYWORD_FOUND (BIT) PAA 283
END_OF_DATA 251
END_OF_DATA 4 L2BL 211
END_OF_DATA 4 L2CH 224
END_OF_MESSAGE 1 MEMMS 256
END_OF_MODULE 1 MEMMS 256
END_OF_SYMSTRING 1 MEMMS 256
ENDREQ_XC (BIT) CCGD 30
enf
  domain manager enf state, DMENC 52
ENF_ANCHOR (0) DMENC 52
ENF_ANCHOR_ADDRESS 47
ENF_ANCHOR_EYE (2) DMENC 52
ENF_ANCHOR_LENGTH (0) DMENC 52
ENF_ELEM (0) DMENC 53
ENF_ELEM_CODE (14) DMENC 53
ENF_ELEM_EYE (2) DMENC 53
ENF_ELEM_LENGTH (0) DMENC 53
ENF_ELEM_LISTENER (10) DMENC 53
ENF_ELEM_NEXT (10) DMENC 53
ENF_EVENT_ARRAY 52
ENF_EVENT_ARRAY_LISTENER (20) DMENC 52
ENF_EVENT_ARRAY_TIME 52
ENF_LISTEN_ELEM (0) DMENC 52
ENF_LISTEN_ELEM_CODE (14) DMENC 52
ENF_LISTEN_ELEM_DELETED (BIT) DMENC 52
ENF_LISTEN_ELEM_DOMAIN (18) DMENC 52
ENF_LISTEN_ELEM_EYE (2) DMENC 52
ENF_LISTEN_ELEM_GATE (1C) DMENC 52
ENF_LISTEN_ELEM_LENGTH (0) DMENC 52
ENF_LISTEN_ELEM_NEXT (10) DMENC 52
ENF_PRIVATE_QUEUE (14) DMENC 52
ENF_PUBLIC_QUEUE (10) DMENC 52
ENF_WAKEUP_ECB (18) DMENC 52
ENF_WAKEUP_ECB_POSTED (BIT) DMENC 52
ENQ_DEQ_ERROR_CODE 4 LGANC 193
enqueue
  enqueue domain anchor block, NQA 275
  enqueue domain browse element, NQB 276
  enqueue domain browse owner extension, NQOX 279
  enqueue domain browse waiter extension, NQWX 282
  enqueue domain enqueue pool, NQPL 280
  enqueue domain queue element area, NQEA 277
ENQUEUE_TIME (50) DSTSK 65
ENQUEUE_TIME_IN_SECS (50) DSTSK 65
ENT (0) D2ENT 81
entry
  kernel stack entry, KESTP 163
  partner table entry, PTE 297
ENVIRONMENT (2C) CCGD 29
EOD (BIT) STUCB 376
ERB (0) DUFC 76
ERB_INDEX (4) DUFC 76
ERB_NEXT (0) DUFC 76
ERB_PAGE_NUMBER (8) DUFC 76
ERDSA 4 SMDCC 363
ERDSA_NAME 5 LDCBS 175
ERDSA_NAME 8 SMDCC 363
ERGN_NAME 5 LDCBS 175
ERH_ARROW (2) KECB 158
ERH_BLOCK_NAME (8) KECB 159
ERH_DFH (3) KECB 158
ERH_DOMID (6) KECB 159
ERH_ENTRY_LENGTH (18) KECB 159
ERH_FIRST_FREE (20) KECB 159
ERH_GUARD (24) KECB 159
ERH_LENGTH (0) KECB 158
ERH_PREFIX (0) KECB 158
ERH_QUICK_CELL (20) KECB 159
ERH_TABLE_END (14) KECB 159
ERH_TABLE_START (10) KECB 159
ERRA_PTR (28) PAA 283
error
  web error program parms, WBEPC 419
ERROR_DIRECTION (28) CPCPS 32
ERROR_ENTRY (28) KECB 159
ERROR_ENTRY_NUMBER 4 KECB 161
ERROR_HEADER (0) KECB 158
ERROR_TABLE (0) KECB 158
ES_DISABLED 1 BAPT 24
ES_ENABLED 1 BAPT 24
ESDSA 4 SMDCC 363
ESDSA_NAME 5 LDCBS 175
ESDSA_NAME 8 SMDCC 363
ESSENTIAL_TCB (BIT) DSANC 57, 59, 60
ESTAE_WAITERS_ECB (D8) DSANC 59
EUDSA 4 SMDCC 363
EUDSA_NAME 8 SMDCC 363
EVENT (F0) BAACT 16
EVENT_POOL_TOKEN (10) BAACT 10
EVENT_VERSION (100) BAACT 16
EXCEPTION_ADDRESS (27C) APLI 4
EXCEPTION_LIST_ADDR (30) SOA 371
EXCEPTION_LIST_LENGTH (2C) SOA 371

EXEC_ASYNCHRONOUS 1 BAACT 20
EXEC_CAPABLE (BIT) DSANC 57, 60
EXEC_MODE (0) BAACT 16
EXEC_SYNCHRONOUS 1 BAACT 20
EXECUTABLE_CHAIN (AC) DSANC 55
EXECUTABLE_CHAIN_LOCK (90) DSANC 55
EXECUTABLE_HEADER (AC) DSANC 55
EXECUTABLE_NEXT 65
EXISTENCE_LOCKED 312, 331
EXISTENCE_LOG_RECORD 4 RMUW 336, 340
EXISTENCE_TO_BE_LOGGED (BIT) RMLK 312
EXISTENCE_TO_BE_LOGGED (BIT) RMUW 331
EXIT_ERROR_ENTRY (20) CAUTR 27
EXIT_ERROR_MODULE (0) CAUTR 28
EXIT_ERROR_MODULE (20) CAUTR 27
EXIT_ERROR_SPACE (25) CAUTR 27
EXIT_ERROR_SPACE (5) CAUTR 28
EXIT_ERROR_TASKNUM (2C) CAUTR 27
EXIT_ERROR_TASKNUM (C) CAUTR 28
EXIT_ERROR_TEXT (26) CAUTR 27
EXIT_ERROR_TEXT (6) CAUTR 28
EXIT_ERROR_TIME 27, 28
EXIT_ERROR_TM_FUNCTION (14) CAUTR 28
EXIT_ERROR_TM_FUNCTION (34) CAUTR 27
EXIT_ERROR_TM_REASON (16) CAUTR 28
EXIT_ERROR_TM_REASON (36) CAUTR 27
EXIT_ERROR_TM_RESPONSE (15) CAUTR 28
EXIT_ERROR_TM_RESPONSE (35) CAUTR 27
EXIT_ERROR_TM_TABLE (10) CAUTR 28
EXIT_ERROR_TM_TABLE (30) CAUTR 27
EXIT_EVENT_ENTRY (20) CAUTR 27
EXIT_EVENT_MODULE (0) CAUTR 27
EXIT_EVENT_MODULE (20) CAUTR 27
EXIT_EVENT_SPACE (25) CAUTR 27
EXIT_EVENT_SPACE (5) CAUTR 27
EXIT_EVENT_TASKNUM (14) CAUTR 28
EXIT_EVENT_TASKNUM (34) CAUTR 27
EXIT_EVENT_TEXT (26) CAUTR 27
EXIT_EVENT_TEXT (6) CAUTR 28
EXIT_EVENT_TIME (18) CAUTR 28
EXIT_EVENT_TIME (38) CAUTR 27
EXITERRORENTRY (0) CAUTR 28
EXITERRORTEXT (0) CAUTR 27
EXITEVENTENTRY (0) CAUTR 27
EXITEVENTTEXT (0) CAUTR 27
EXPIRATION_TOKEN (148) DSANC 56
EXT_CHEAPEXIT (18) DSTSK 67
EXT_ENTRY_TAB_PTR (8D0) STUCB 376
EXT_MODE (9) DSTSK 67
EXT_POSTEXIT (C) DSTSK 67
EXT_RES (A) DSTSK 67
EXT_ST_EXIT_RAN 4 DSTSK 68
EXT_ST_EXT_COMPL 4 DSTSK 68
EXT_ST_EXTEND 4 DSTSK 68
EXT_ST_UNUSED 4 DSTSK 68
EXT_STATUS (14) DSTSK 67
EXT_THISTASK (10) DSTSK 67
EXT_USER (10) DSTSK 67
EXT_VALUE (8) DSTSK 67
extended
  logger reusable extended iliffe vector class, RUEI 343
extension
  device support extension, FEP08 127
  enqueue domain browse owner extension, NQOX 279
  enqueue domain browse waiter extension, NQWX 282
EXTENSION (0) DSTSK 67
EXTENSION_ADDRESS (60) DSTSK 65
EXTENSION_CELL_ROOT (F0) DSANC 55
EXTENSION_PAGE_MAP (10) DSANC 62
EXTENSIONS_IN_BLOCK 4 DSTSK 68
external
  external CICS interface control blocks, XCCBC 431
eye
  tsf - eye catcher map, FEP09 131
EYE_CATCHER (0) BAACT 21
EYE_CATCHER (0) BAPT 23
EYE_CATCHER (0) CAUTR 26
EYE_CATCHER (0) DSANC 53, 58, 61, 62
EYE_CATCHER (0) L2SL 240
EYE_CATCHER (0) RMNS 323
EYE_CATCHER (100) RMUW 339
EYE_CATCHER (40) RMLK 305
EYE_CATCHER (460) RMLK 306
EYE_CATCHER (520) RMUW 339
EYE_CATCHER (8) L2BL 208
EYE_CATCHER (8) L2BS 212
EYE_CATCHER (8) L2CH 220
EYE_CATCHER (8) L2HS 229
EYE_CATCHER (8) L2SR 243
EYE_CATCHER (8) RMLK 305, 309
EYE_CATCHER (910) RMLK 307
EYE_CATCHER (F8) L2BS 215
EYE_CATCHER (F8) L2SR 247
EYE_CATCHER 14 BAPT 24
EYE_LEN (0) BAACT 5, 10, 16, 21
EYE_LEN (0) BAPT 23
EYE_LEN (0) CAUTR 26, 27
EYE_LEN (10) BAACT 17
EYE_LT (4) CAUTR 27
EYE_NAME (8) CAUTR 27
EYE_OFFSET (12) BAACT 17
EYE_OFFSET (2) BAACT 5, 10, 16, 22
EYE_OFFSET (2) BAPT 23
EYE_OFFSET (2) CAUTR 26, 27
EYE_PFX (5) CAUTR 27
EYE_STRING (14) BAACT 17
EYE_STRING (4) BAACT 5, 10, 16, 22
EYE_STRING (4) BAPT 23
EYE_STRING (4) CAUTR 26, 27
EYECATCHER (0) CAUTR 27
EYECATCHER_ARROW 1 LDCBS 174
EYECATCHER_ARROW 1 MNCBS 273
EYECATCHER_DFH 3 LDCBS 174
EYECATCHER_DFH 3 MNCBS 273
EYECATCHER_DOMID 2 LDCBS 174
EYECATCHER_DOMID 2 MNCBS 273

F

facility
  dm authorised facility state, DMAFC 45
  terminal simulation facility, FEP19 146
failure
  log of logs failure record, LGFL 198
FAILURE_TIME (60) RMLK 310
FAILURE_TIME (968) RMLK 308
FAILURE_TIME (C) RMLK 317
FALSE 391
FALSE 0 CCGD 31
FALSE 0 DDCBC 38
FALSE 0 STUCB 378
FASTPATH_FLAGS 293
FBWA (0) FBWAC 100
FBWA_BACKWARDS (BIT) FBWAC 100
FBWA_CURRENT_KEY (18) FBWAC 100
FBWA_EYE_CATCHER (0) FBWAC 100
FBWA_EYE1 (2) FBWAC 100
FBWA_EYE2 (8) FBWAC 100
FBWA_FIRST (BIT) FBWAC 100
FBWA_FIXED_END (30) FBWAC 100
FBWA_FIXED_PART (0) FBWAC 100
FBWA_FLAGS1 (10) FBWAC 100
FBWA_FLAGS2 100
FBWA_FREE_CHAIN (14) FBWAC 100
FBWA_GENERIC (BIT) FBWAC 100
FBWA_GTEQ (BIT) FBWAC 100
FBWA_KEY_LENGTH 100
FBWA_KEYS (30) FBWAC 100
FBWA_LENGTH (0) FBWAC 100
FBWA_NEXT_KEY (20) FBWAC 100
FBWA_NEXT_KEY_VALID (BIT) FBWAC 100
FBWA_RBA (BIT) FBWAC 100
FBWA_RECORD_TOKEN (24) FBWAC 100
FBWA_REQUEST_KEY (1C) FBWAC 100
FBWA_SEQUENTIAL (BIT) FBWAC 100
FBWA_SOURCE_CURRENT (BIT) FBWAC 100
FBWA_SOURCE_IN_SEQ (BIT) FBWAC 100
FBWA_SOURCE_STARTED (BIT) FBWAC 100
FBWA_TOKEN_VALID (BIT) FBWAC 100
FBWAC 99
FCPE_CONNECT_FAILED (BIT) FCPEC 101
FCPE_CONNECT_IN_PROGRESS (BIT) FCPEC 101
FCPE_CONNECTION_TOKEN (20) FCPEC 101
FCPE_COUNT_OF_OPENS (24) FCPEC 101
FCPE_EYE_CATCHER (0) FCPEC 101

FCPE_EYE1 (2) FCPEC 101
FCPE_EYE2 (8) FCPEC 101
FCPE_FIRST_LRS_WAITER (38) FCPEC 102
FCPE_FIRST_WAITER (40) FCPEC 102
FCPE_FLAGS (2C) FCPEC 101
FCPE_INSTANCE_NUMBER (28) FCPEC 101
FCPE_LAST_LRS_WAITER (3C) FCPEC 102
FCPE_LAST_WAITER (44) FCPEC 102
FCPE_LENGTH (0) FCPEC 101
FCPE_LOCK_TOKEN (30) FCPEC 102
FCPE_LRS_COUNT (34) FCPEC 102
FCPE_LRS_WAIT_HEAD (38) FCPEC 102
FCPE_MAIN_PART (10) FCPEC 101
FCPE_NEXT_ADDRESS (10) FCPEC 101
FCPE_OPEN_FILE_CHAIN (48) FCPEC 102
FCPE_POOL_NAME (18) FCPEC 101
FCPE_PREV_ADDRESS (14) FCPEC 101
FCPE_RESTARTED (BIT) FCPEC 101
FCPE_WAIT_HEAD (40) FCPEC 102
FCPEC 101
FCPW_CHAIN (10) FCPWC 103
FCPW_EYE_CATCHER (0) FCPWC 103
FCPW_EYE1 (2) FCPWC 103
FCPW_EYE2 (8) FCPWC 103
FCPW_FLAGS (29) FCPWC 104
FCPW_LENGTH (0) FCPWC 103
FCPW_LRS_WAIT (BIT) FCPWC 104
FCPW_MAIN_PART (10) FCPWC 103
FCPW_MAXREQS_WAIT (BIT) FCPWC 104
FCPW_NEXT_ADDRESS (10) FCPWC 103
FCPW_PREV_ADDRESS (14) FCPWC 103
FCPW_RESUME_PRIORITY (28) FCPWC 104
FCPW_SUSPEND_TIME (20) FCPWC 103
FCPW_SUSPEND_TOKEN (18) FCPWC 103
FCPW_TASK_TOKEN (1C) FCPWC 103
FCPW_TRAN_NUM (2C) FCPWC 104
FCPWC 102
FCQRE 104
FCQRE_ARROW (2) FCQRE 104
FCQRE_BLOCKNAME (8) FCQRE 104
FCQRE_BODY (18) FCQRE 104
FCQRE_BWO_END 1 FCQRE 105
FCQRE_BWO_START 1 FCQRE 105
FCQRE_CACHE (18) FCQRE 104
FCQRE_CACHE_AVAILABLE 1 FCQRE 105
FCQRE_CACHE_LENGTH (54) FCQRE 105
FCQRE_CONCURRENT (BIT) FCQRE 104
FCQRE_DATASET (18) FCQRE 104
FCQRE_DATASET_LENGTH (54) FCQRE 105
FCQRE_DFH (3) FCQRE 104
FCQRE_DOMAIN (6) FCQRE 104
FCQRE_ELEMENT_TYPE (44) FCQRE 104
FCQRE_ERROR_DATA (50) FCQRE 104
FCQRE_ERROR_REQUEST 1 FCQRE 105
FCQRE_ERROR_TYPE (46) FCQRE 104
FCQRE_ERROR_USED (BIT) FCQRE 104
FCQRE_EYE 8 FCQRE 105
FCQRE_FLAGS (47) FCQRE 104
FCQRE_FWD_RECOV_COMPLETE 1 FCQRE 105
FCQRE_IMMEDIATE (BIT) FCQRE 104
FCQRE_LENGTH (0) FCQRE 104
FCQRE_LOCKS_RECOV_COMPLETE 1 FCQRE 105
FCQRE_NEXT (10) FCQRE 104
FCQRE_NEXT_ISOLATE (14) FCQRE 104
FCQRE_NONBWO_END 1 FCQRE 105
FCQRE_NONBWO_START 1 FCQRE 105
FCQRE_PREFIX (0) FCQRE 104
FCQRE_QUICMP_TOKEN 104
FCQRE_QUIESCE 1 FCQRE 105
FCQRE_QUIESCE_REQUEST 1 FCQRE 105
FCQRE_QUIESCE_TYPE (45) FCQRE 104
FCQRE_STG_FAILURE 1 FCQRE 105
FCQRE_UNQUIESCE 1 FCQRE 105
FCQSE 105
FCQSE_ARROW (2) FCQSE 105
FCQSE_BLOCKNAME (8) FCQSE 105
FCQSE_BODY (18) FCQSE 105
FCQSE_BWO_CANCEL 1 FCQSE 106
FCQSE_CANCELLED 1 FCQSE 106
FCQSE_CICS (BIT) FCQSE 105
FCQSE_CONF_BWO 1 FCQSE 106
FCQSE_CONF_NONBWO 1 FCQSE 106
FCQSE_CONF_QUIESCE 1 FCQSE 106
FCQSE_CONF_UNKNOWN 1 FCQSE 106
FCQSE_CONF_UNQUIESCE 1 FCQSE 106
FCQSE_CONFLICT (54) FCQSE 106
FCQSE_DATASET_MIGRATED 1 FCQSE 106
FCQSE_DFH (3) FCQSE 105
FCQSE_DOMAIN (6) FCQSE 105
FCQSE_DSNAME (18) FCQSE 105
FCQSE_DSNAME_LENGTH (68) FCQSE 106
FCQSE_EYE 8 FCQSE 106
FCQSE_FLAGS (45) FCQSE 105
FCQSE_IMMQUIESCE 1 FCQSE 106
FCQSE_IOERR 1 FCQSE 106
FCQSE_LENGTH (0) FCQSE 105
FCQSE_NEW_STATE 1 FCQSE 106
FCQSE_NEXT (10) FCQSE 105
FCQSE_NONBWO_CANCEL 1 FCQSE 106
FCQSE_OK 1 FCQSE 106
FCQSE_PREFIX (0) FCQSE 105
FCQSE_PREV (14) FCQSE 105
FCQSE_QUIESCE 1 FCQSE 106
FCQSE_QUIESCE_CANCEL 1 FCQSE 106
FCQSE_QUIESCE_NOT_POSSIBLE 1 FCQSE 106
FCQSE_QUIESCE_TYPE (44) FCQSE 105
FCQSE_R15 (62) FCQSE 106
FCQSE_REASON (63) FCQSE 106
FCQSE_RESP_CODE 106
FCQSE_RESUMED_STATE 1 FCQSE 106
FCQSE_SENT_STATE 1 FCQSE 106
FCQSE_SERVER_FAILURE 1 FCQSE 106
FCQSE_STATE (47) FCQSE 106
FCQSE_SUSPEND_TOKEN (48) FCQSE 106
FCQSE_TIMED_OUT 1 FCQSE 106
FCQSE_TIMEDOUT_STATE 1 FCQSE 106
FCQSE_TIMEOUT_TIME (50) FCQSE 106
FCQSE_TRAN_NUMBER (64) FCQSE 106
FCQSE_UNKNOWN_VSAM_DATASET 1 FCQSE 106
FCQSE_UNQUIESCE 1 FCQSE 106
FCQSE_UNQUIESCE_NOT_POSSIBLE 1 FCQSE 106
FCQSE_USER_NOT_AUTH 1 FCQSE 106
FCQSE_USERID 106
FCQSE_VSAM_ECB_ADDR (4C) FCQSE 106
FCQSE_VSAM_ERROR 1 FCQSE 106
FCQSE_VSAM_RC (62) FCQSE 106
FCQSE_WAIT (BIT) FCQSE 105
FCUP_CHAIN (10) FCUPC 107
FCUP_EYE_CATCHER (0) FCUPC 107
FCUP_EYE1 (2) FCUPC 107
FCUP_EYE2 (8) FCUPC 107
FCUP_FRAB_PTR (28) FCUPC 107
FCUP_LENGTH (0) FCUPC 107
FCUP_LINK_TOK (20) FCUPC 107
FCUP_MAIN_PART (10) FCUPC 107
FCUP_NEXT_ADDRESS (10) FCUPC 107
FCUP_POOL_ELEM_PTR (24) FCUPC 107
FCUP_POOL_NAME (18) FCUPC 107
FCUP_PREV_ADDRESS (14) FCUPC 107
FCUPC 107
FEATURE_DEFAULT_LANG_PTR (118) MEPS 257
FEATURE_MSG_MOD_PTRS (120) MEPS 257
FEP01 108
FEP02 113
FEP03 115
FEP04 116
FEP05 117
FEP06 120
FEP07 125
FEP08 127
FEP09 131
FEP10 132
FEP11 134
FEP12 135
FEP13 136
FEP14 138
FEP15 139
FEP16 140
FEP17 141
FEP18 145
FEP19 146
FEP20 147
FEP21 148
file
    file browse work area for data tables, FBWAC 99

file *(continued)*
  file control cfdt pool element, FCPEC 101
  file control cfdt pool wait element, FCPWC 102
  file control cfdt uow pool block, FCUPC 107
  file control locks locator block, FLLBC 150
  file control quiesce receive element, FCQRE 104
  file control quiesce send element, FCQSE 105
FILE (8) BAPT 23
FILE_CLOSED 1 CCGD 31
FILE_DESCRIPTOR (18) SOA 372
FILE_OPEN 1 CCGD 31
FILENAME (4) BAACT 12, 18
FILENAME (8) BAACT 6, 7
FILL (34) CPCPS 32
FINISH (D0) L2CH 223
FIRE_REQUEST 1 BAACT 20
FIRST_BLOCK (38) L2BS 212
FIRST_BLOCK (38) L2SR 243
FIRST_COMMIT_DONE 312, 331
FIRST_CONVERS (BIT) XCCBC 433
FIRST_INPUT_RECORD (BIT) STUCB 376
FIRST_OUTPUT_RECORD (BIT) STUCB 376
FIRST_POOL (40) PAA 283
FIRST_REC (48) PAA 283
FIRST_STIMER (10) DSANC 61
FIRST_TIMEOUT 59
FIRST_UOW_FOR_TRANSACTION 311, 331
FIXED_LENGTH_MAXIMUM 4 TSMN 393
FIXED_LENGTH_MULTIPLE 4 TSMN 393
FIXED_SUBPOOLS 4 TSMN 393
FLAGS (20) RMLK 316
FLAGS (3C) L2CH 220
FLAGS (40) RMNS 323
FLAGS (58) RMLK 311
FLAGS (58) RMUW 331
FLAGS (90) RMNM 321
FLAT_ACTIVITY_LENGTH 4 BAACT 21
FLAT_ACTIVITY_SPARE 4 BAACT 21
FLAT_BLOCK (0) L2LF 231, 233
FLAT_BLOCK (10) L2LF 235
FLAT_BLOCK (10) LGSF 200
FLAT_BLOCK (14) L2LF 234
FLAT_BLOCK (24) L2LF 235, 236
FLAT_BLOCK (24) LGSF 200, 201
FLAT_BLOCK (34) L2LF 235
FLAT_BLOCK (34) LGSF 201
FLAT_BLOCK (4) L2LF 234, 235
FLAT_BLOCK_ID (0) L2LF 231, 233
FLAT_BLOCK_ID (10) L2LF 235
FLAT_BLOCK_ID (10) LGSF 200
FLAT_BLOCK_ID (14) L2LF 234
FLAT_BLOCK_ID (24) L2LF 235, 236
FLAT_BLOCK_ID (24) LGSF 200, 201
FLAT_BLOCK_ID (34) L2LF 235
FLAT_BLOCK_ID (34) LGSF 201
FLAT_BLOCK_ID (4) L2LF 234, 235
FLAT_BLOCK_NUM (0) L2LF 231, 233
FLAT_BLOCK_NUM (10) L2LF 235
FLAT_BLOCK_NUM (10) LGSF 200
FLAT_BLOCK_NUM (14) L2LF 234
FLAT_BLOCK_NUM (24) L2LF 235, 236
FLAT_BLOCK_NUM (24) LGSF 200, 201
FLAT_BLOCK_NUM (34) L2LF 235
FLAT_BLOCK_NUM (34) LGSF 201
FLAT_BLOCK_NUM (4) L2LF 234, 235
FLAT_EPOOL_LEN (A0) BAACT 18
FLAT_EPOOL_LEN (C0) BAACT 11
FLAT_EPOOL_PTR (9C) BAACT 18
FLAT_EPOOL_PTR (BC) BAACT 11
FLAT_INDEX (10) L2LF 234, 235
FLAT_INDEX (1C) L2LF 235
FLAT_INDEX (1C) LGSF 200
FLAT_INDEX (20) L2LF 234
FLAT_INDEX (30) L2LF 235, 236
FLAT_INDEX (30) LGSF 200, 201
FLAT_INDEX (40) L2LF 235
FLAT_INDEX (40) LGSF 201
FLAT_INDEX (C) L2LF 231, 233
FLAT_PROCESS_LENGTH 4 BAACT 8
FLAT_PROCESS_SPARE 4 BAACT 8
FLAT_REAL (19) L2LF 235
FLAT_REAL (19) LGSF 200
FLAT_REAL (1D) L2LF 234
FLAT_REAL (2D) L2LF 235, 236
FLAT_REAL (2D) LGSF 200, 201
FLAT_REAL (3D) L2LF 235
FLAT_REAL (3D) LGSF 201
FLAT_REAL (9) L2LF 231, 233
FLAT_REAL (D) L2LF 234, 235
FLAT_RSVD1 (1A) L2LF 235
FLAT_RSVD1 (1A) LGSF 200
FLAT_RSVD1 (1E) L2LF 234
FLAT_RSVD1 (2E) L2LF 235, 236
FLAT_RSVD1 (2E) LGSF 200, 201
FLAT_RSVD1 (3E) L2LF 235
FLAT_RSVD1 (3E) LGSF 201
FLAT_RSVD1 (A) L2LF 231, 233
FLAT_RSVD1 (E) L2LF 234, 235
FLAT_SET_ELEMENT_LENGTH 4 BAACT 20
FLAT_SET_ELEMENT_SPACE 14
FLATBLOCK (0) L2LF 231
FLATRECORDTOKEN (0) L2LF 231
FLLB_DSNB_ADDRESS (10) FLLBC 150
FLLB_EYE_CATCHER (0) FLLBC 150
FLLB_EYE1 (2) FLLBC 150
FLLB_EYE2 (8) FLLBC 150
FLLB_LENGTH (0) FLLBC 150
FLLB_LOCK_CONDITION (28) FLLBC 150
FLLB_LOST_LOCKS (BIT) FLLBC 150
FLLB_LUWID (20) FLLBC 150
FLLB_MAIN_PART (10) FLLBC 150
FLLB_NEXT_IN_DSNB_CHAIN (14) FLLBC 150
FLLB_NEXT_IN_FRAB_CHAIN (1C) FLLBC 150
FLLB_OFFSITE_RECOVERY (BIT) FLLBC 151
FLLB_OVERRIDEN_LOCKS (BIT) FLLBC 150
FLLB_PREV_IN_DSNB_CHAIN (18) FLLBC 150
FLLBC 150
FLOATING_POINT_REG0 (15C) APLI 4
FLOATING_POINT_REG2 (164) APLI 4
FLOATING_POINT_REG4 (16C) APLI 4
FLOATING_POINT_REG6 (174) APLI 4
FLOATING_POINT_REGISTERS (15C) APLI 3
FLUSHED 1 L2SR 251
FORCE_PURGE_PROTECTION (BIT) RMLK 312
FORCE_PURGE_PROTECTION (BIT) RMUW 331
FORCE_TOKEN (3C) L2BS 212
FORCE_TOKEN (3C) L2SR 244
FORCE_TOKEN (4) L2SR 250
FORCE_TOKEN (7C) L2BS 213
FORCE_TOKEN (7C) L2SR 244
FORCE_WAITS_CU (224) L2BS 216
FORCE_WAITS_CU (224) L2SR 248
FORCE_WAITS_PK (228) L2BS 216
FORCE_WAITS_PK (228) L2SR 248
FORCE_WAITS_TO (22C) L2BS 216
FORCE_WAITS_TO (22C) L2SR 248
FORGET (19) RMLK 317
FORGET (6D) RMLK 310
FORGET (975) RMLK 308
FORK_CHAIN_HEADER (0) L2LF 234
FORK_RM_START (24) L2LF 234
FORK_RM_START (44) L2LF 235
FORK_RM_START (44) LGSF 201
format
  system log format, LGSF 199
FORMAT_CHAR 1 MEMMS 256
FORMAT_DATE 1 MEMMS 256
FORMAT_DEC 1 MEMMS 256
FORMAT_HEX 1 MEMMS 256
FORMAT_OPT 1 MEMMS 256
FORMAT_TIME 1 MEMMS 256
formats
  log manager log formats, L2LF 231
FORMATTER_FLAGS (92E) STUCB 377
formatting
  dump formatting communication area, DUFC 75
FREE_1_NEXT (0) LMCB2 207
FREE_2_NEXT (0) LMCB2 207
FREE_3_NEXT (0) LMCB2 207
FREE_CHAIN_CDS (B8) DSANC 55
FREE_CHAIN_CDS (C8) DSANC 55
FREE_CHAIN_CDS (D8) DSANC 55
FREE_CHAIN_CDS (E8) DSANC 55
FREE_CHAIN_CDS (F8) DSANC 55
FREE_CHAIN_COUNT (BC) DSANC 55
FREE_CHAIN_COUNT (CC) DSANC 55

FREE_CHAIN_COUNT (DC) DSANC 55
FREE_CHAIN_COUNT (EC) DSANC 55
FREE_CHAIN_COUNT (FC) DSANC 55
FREE_CHAIN_HEAD (114) RMUW 339
FREE_CHAIN_HEAD (474) RMLK 306
FREE_CHAIN_HEAD (534) RMUW 339
FREE_CHAIN_HEAD (54) RMLK 305
FREE_CHAIN_PTR (B8) DSANC 55
FREE_CHAIN_PTR (C8) DSANC 55
FREE_CHAIN_PTR (D8) DSANC 55
FREE_CHAIN_PTR (E8) DSANC 55
FREE_CHAIN_PTR (F8) DSANC 55
FREE_CHAINS 58
FREE_DS_TCBS (690) DSANC 58
FREE_HEADER (BIT) BAACT 21
FREE_OPEN_BASESPACE_DS_TCBS (678) DSANC 58
FREE_OPEN_SUBSPACE_DS_TCBS (684) DSANC 58
FREECHAIN_1 (0) LMCB2 207
FREECHAIN_2 (0) LMCB2 207
FREECHAIN_3 (0) LMCB2 207
FRONT_PTR (0) DSANC 60
FRONT_PTR (1C) DSANC 58
frontend
    frontend programming interface trace, FEP01 108
    frontend programming interface, FEP21 148
FRST (10) DDBSC 35
FUNCTIONCODE (18) SOA 372

G

GC_LOCK 8 CCGD 31
GEN_INSERT_LEN (4) MEPS 258
GEN_INSERT_PTR (0) MEPS 258
GENERAL_FLAGS (E0) DSTSK 66
GENERAL_INSERT (0) MEPS 258
GENERAL_NEXT (30) DSTSK 65
GENERATION (A4) BAACT 18
GENERATION (C4) BAACT 11
GENERIC_LAI (45) RMUW 335
GENLOGRECORD (0) L2LF 233
GENLOGUSER (0) L2LF 236
GETCLIENTID_PARMS (18) SOA 372
GETFLAG 1 SMMCC 366
GETFLAG_OFF 1 SMMCC 366
GETHOSTNAME_PARMS (18) SOA 372
GETPAGE_LOCK (5E0) DSANC 57
GIVESOCKET_PARMS (18) SOA 372
GLB_ATTACH_DETACH_CHAIN (CC) D2GLB 87
GLB_ATTACH_PARMLIST (12C) D2GLB 87
GLB_ATTACH_STATUS 86
GLB_CICS_CHAPPED_DOWN (BIT) D2GLB 87
GLB_CICS_ID (18) D2GLB 85
GLB_COMD (488) D2GLB 89
GLB_CONNECT_ERROR 86
GLB_CONNECT_ERROR_ABEND (BIT) D2GLB 86
GLB_CONNECT_ERROR_SQLCODE (BIT) D2GLB 86
GLB_CONNECT_TIME (5C) D2GLB 85
GLB_CONNECTED (BIT) D2GLB 86
GLB_CONNECTING (BIT) D2GLB 86
GLB_CONNECTION_STATUS 86
GLB_CURRENT_TCBS (78) D2GLB 85
GLB_DB2_ACCMAINT (BIT) D2GLB 86
GLB_DB2_ID (20) D2GLB 85
GLB_DB2_RELEASE (24) D2GLB 85
GLB_DB2CONN_NAME (10) D2GLB 85
GLB_DFHD2EX1_GWA_ADDR (2C) D2GLB 85
GLB_DFHD2EX2_ENTRY (30) D2GLB 85
GLB_DFHD2EX3_ENTRY (34) D2GLB 85
GLB_DFHD2MSB_ACTIVE (BIT) D2GLB 86
GLB_DFHD2MSB_ENTRY (38) D2GLB 85
GLB_DISCARDING_DB2CONN (BIT) D2GLB 86
GLB_DISCONNECT_TIME (64) D2GLB 85
GLB_DISCONNECTING (BIT) D2GLB 86
GLB_DSNAPRH_ENTRY (28) D2GLB 85
GLB_EYE (2) D2GLB 85
GLB_FLAGS (A8) D2GLB 86
GLB_FRB (23C) D2GLB 87
GLB_FREE_PROT_THREAD_CHAIN1 (A0) D2GLB 86
GLB_FREE_PROT_THREAD_CHAIN2 (A4) D2GLB 86
GLB_FREE_TCB_CHAIN (84) D2GLB 85
GLB_FREE_TCB_COUNT (88) D2GLB 85
GLB_IDENTIFY_TERMINATED (BIT) D2GLB 87
GLB_IN_STANDBY (BIT) D2GLB 86
GLB_INDOUBT_LIST (C0) D2GLB 87
GLB_INDOUBTS_COUNT (C6) D2GLB 87
GLB_INDOUBTS_LENGTH (C4) D2GLB 87
GLB_LEN (0) D2GLB 85
GLB_MSB_ABENDING (BIT) D2GLB 87
GLB_MSB_AREA (C0) D2GLB 87
GLB_MSB_DB2_IDENTIFY_FAILED (BIT) D2GLB 87
GLB_MSB_DB2_NOT_ACTIVE (BIT) D2GLB 87
GLB_MSB_EST_ESTAE_FAILED (BIT) D2GLB 87
GLB_MSB_EST_EXIT_FAILED (BIT) D2GLB 87
GLB_MSB_INSUFFICIENT_AUTH (BIT) D2GLB 87
GLB_MSB_ISSUED_ABEND (BIT) D2GLB 87
GLB_MSB_LISTEN_ECB (D4) D2GLB 87
GLB_MSB_LOAD_PRH_FAILED (BIT) D2GLB 87
GLB_MSB_PARM2 (E2) D2GLB 87
GLB_MSB_PARM3 (E1) D2GLB 87
GLB_MSB_PARM4 (E0) D2GLB 87
GLB_MSB_SAVEAREA (E4) D2GLB 87
GLB_MSB_SHOW_INDOUBT_FAILED (BIT) D2GLB 87
GLB_MSB_START_ECB (D8) D2GLB 87
GLB_MSB_STOP_ECB (DC) D2GLB 87
GLB_MSB_TCB (3C) D2GLB 85
GLB_MSB_WAIT_ECB (D0) D2GLB 87
GLB_MSG_QUEUE1 (40) D2GLB 85
GLB_MSG_QUEUE2 (44) D2GLB 85
GLB_MSG_QUEUE3 (48) D2GLB 85
GLB_MSG_QUEUES (40) D2GLB 85
GLB_NON_TERMINAL_RELEASE 86
GLB_NON_TERMINAL_RELEASE_YES (BIT) D2GLB 86
GLB_POOL (3C0) D2GLB 88
GLB_PREFIX (0) D2GLB 85
GLB_PURGE_CYCLE (70) D2GLB 85
GLB_PURGE_CYCLE_MINUTES (70) D2GLB 85
GLB_PURGE_CYCLE_SECONDS (74) D2GLB 85
GLB_SDWA_ADDRESS (3BC) D2GLB 88
GLB_SDWA_NAME (3B4) D2GLB 88
GLB_SDWA_PSW (3AC) D2GLB 88
GLB_SDWA_REGS (36C) D2GLB 88
GLB_SECURITY_REBUILD_TIME (54) D2GLB 85
GLB_SERVICE_TASK_ECB (B0) D2GLB 87
GLB_SERVICE_TASK_P_COUNT 87
GLB_SERVICE_TASK_STARTED (BIT) D2GLB 86
GLB_SERVICE_TASK_STOP_ECB (B4) D2GLB 87
GLB_SERVICE_TASK_TERMINATE (BIT) D2GLB 86
GLB_SHUTDOWN_CICS_IMMED (BIT) D2GLB 87
GLB_SHUTDOWN_CICS_QUIESCE (BIT) D2GLB 87
GLB_SHUTDOWN_DB2 (BIT) D2GLB 87
GLB_SHUTDOWN_EX1_FINAL (BIT) D2GLB 87
GLB_SHUTDOWN_EX2 (BIT) D2GLB 87
GLB_SHUTDOWN_FLAGS 86
GLB_SHUTDOWN_FORCE (BIT) D2GLB 86
GLB_SHUTDOWN_MSB_ESTAE (BIT) D2GLB 87
GLB_SHUTDOWN_QUIESCE (BIT) D2GLB 86
GLB_SIGNON_ID (4C) D2GLB 85
GLB_STANDBY_MODE 86
GLB_STANDBY_MODE_CONNECT (BIT) D2GLB 86
GLB_STANDBY_MODE_NOCONNECT (BIT) D2GLB 86
GLB_STANDBY_MODE_RECONNECT (BIT) D2GLB 86
GLB_STATS_BUFFER_ADDR (368) D2GLB 88
GLB_STATS_BUFFER_LEN 4 D2GLB 91
GLB_STATS_QUEUE (6C) D2GLB 85
GLB_TCB_HWM (7C) D2GLB 85
GLB_TCB_LIMIT (80) D2GLB 85
GLB_TCB_READYQ 85
GLB_TCB_READYQ_CHAIN (90) D2GLB 86
GLB_TCB_READYQ_COUNT (98) D2GLB 86
GLB_TCB_READYQ_COUNTS (98) D2GLB 86
GLB_TCB_READYQ_HWM (9C) D2GLB 86
GLB_TCB_READYQ_SEC_COUNT (94) D2GLB 86
GLB_TCBS (78) D2GLB 85
GLB_TERMINATE_IDENTIFY (BIT) D2GLB 87
GLB_THREAD_ERROR 86
GLB_THREAD_ERROR_ABEND (BIT) D2GLB 86
GLB_THREAD_ERROR_N906 (BIT) D2GLB 86
GLB_THREAD_ERROR_N906D (BIT) D2GLB 86
GLB_THREAD_NUM_WORDS (26C) D2GLB 88
GLB_THREAD_NUMBERS (26C) D2GLB 87
GLB_UR_INDOUBT_LOT_ADDR (C8) D2GLB 87
GLB_WORKAREA (174) D2GLB 87
global
    cics/db2 global block, D2GLB 85

global *(continued)*
  cics/db2 global work area, D2GWA 92
  dce services domain global statistics, DEGPC 38
GLOBAL (0) LDCBS 169
GLOBAL_ARROW (2) LDCBS 169
GLOBAL_BLOCK_ID (8) LDCBS 169
GLOBAL_CATALOG 2 CCGD 31
GLOBAL_CHAIN_LIST (10) L2CH 222
GLOBAL_DFH (3) LDCBS 169
GLOBAL_DOMAIN (6) LDCBS 169
GLOBAL_ID_STRING 8 LDCBS 174
GLOBAL_LENGTH (0) LDCBS 169
GLOBAL_ME 1 CCGD 31
GLOBAL_STREAM_CHAIN (10) L2SR 249
GLRH_GMT (C) L2LF 233
GLRH_HEADER_LENGTH (4) L2LF 233
GLRH_LGSSI (34) L2LF 233
GLRH_LGSSI_FLAGS (34) L2LF 233
GLRH_LGSSI_RSVD (35) L2LF 233
GLRH_LOCAL (14) L2LF 233
GLRH_REC_COMPID (2A) L2LF 233
GLRH_REC_DATA (38) L2LF 233
GLRH_REC_DATA_LEN (8) L2LF 233
GLRH_REC_JOURNAL (2C) L2LF 233
GLRH_REC_TYPE (28) L2LF 233
GLRH_RECORD_ID (28) L2LF 233
GLRH_RECORD_LENGTH 233
GLRH_START_OF_TASK (BIT) L2LF 233
GLRH_START_OF_UOW (BIT) L2LF 233
GLRH_TASK_ID (20) L2LF 233
GLRH_TASK_INFO (1C) L2LF 233
GLRH_TERM_ID (24) L2LF 233
GLRH_TIMESTAMPS (C) L2LF 233
GLRH_TRAN_ID (1C) L2LF 233
GOT_BLOCKS (C3) L2BS 213
GOT_BLOCKS (C3) L2SR 244
GWA_EYE (2) D2GWA 92
GWA_LENGTH (0) D2GWA 92
GWA_LOT (C) D2GWA 92
GWA_OLD_RCT (8) D2GWA 92
GWA_PREFIX (0) D2GWA 92

H

HAND_POST_IGNORE (BIT) DSTSK 65
HAND_POST_NEXT (34) DSTSK 65
HAND_POSTABLE_CHAIN (100) DSANC 56
HAND_POSTABLES (100) DSANC 56
handle
  handle manager declarations, PGHM 293
handler
  document handler anchor block, DHANC 39
  document handler template descriptor, DHTL 43
hard
  log manager hard stream class, L2HS 227
HARD_STREAM (F0) L2BS 213
HARD_STREAM (F0) L2SR 245
HARD_STREAM_PTR (4) L2BL 209
HARDSTREAM (0) L2HS 228
HAS_BEEN_DELETED 308, 310
HAS_BEEN_ISSUE_PREPARED (BIT) RMLK 308, 310
HASHELEM (0) DDCBC 37
HASHSTRUCT (0) DDCBC 37
HDR (0) DDBSC 35
HE_NAME (C) DDCBC 37
HE_NEXT (0) DDCBC 37
HE_TOKEN (4) DDCBC 37
HEAD (4) BAACT 14
HEAD (40) L2CH 221
HEAD (98) BAACT 17
HEAD (B8) BAACT 11
header
  kernel module header, KEMHD 161
  stack segment table header, LIFO 203
headers
  lock manager domain quickcell headers, LMCB2 206
HEARTBEAT_SUSPEND_TOKEN (B0) L2DM 224
HELD (BIT) L2DM 225
HELD (BIT) L2LT 238
HELD (BIT) RMDM 302
HEURISM (52) RMLK 311
HEURISM (52) RMUW 331
HEURISM_FORCED_BY_CLIENT_LU61 (99C) RMUW 340
HEURISM_FORCED_BY_CLIENT_MRO (9A0) RMUW 340
HEURISM_FORCED_BY_CLIENT_OTHER (9A8) RMUW 340
HEURISM_FORCED_BY_CLIENT_RMI (9A4) RMUW 340
HEURISM_FORCED_BY_CLIENT_TD (998) RMUW 340
HEURISM_FORCED_BY_OPERATOR (990) RMUW 340
HEURISM_FORCED_BY_OTHER (994) RMUW 340
HEURISM_FORCED_BY_TIMEOUT (98C) RMUW 340
HEURISM_FORCED_BY_TRANDEF (988) RMUW 340
HEURISTIC_CAUSE (2F) RMLK 311
HEURISTIC_CAUSE (2F) RMUW 330
HEURISTIC_DECISION_TAKEN (BIT) RMLK 312
HEURISTIC_DECISION_TAKEN (BIT) RMUW 331
HIGH_ALLOC_OPEN_TCBS (64C) DSANC 58
HIGH_OPEN_TCBS (654) DSANC 58
history
  log manager history point class, L2HP 226
HISTORY_POINT_INFO 223
HISTORY_POINTS_RESTORED 223
HISTORYPOINT (0) L2HP 226
HOP_FALSE 1 BAACT 10, 19
HOP_TRUE 1 BAACT 10, 19
HOW (18) SOA 372
HP 221
HP_NORMAL 1 L2HP 227
HP_TRIMMED_TO (C1) L2CH 223
HP_ULTIMATE_FUTURE 1 L2HP 227
HP_ULTIMATE_PAST 1 L2HP 227
HPT_LAST_PTR (104) DSANC 56
HPT_WAIT_LIST_CURSOR (110) DSANC 56
HPT_WAIT_LIST_END (10C) DSANC 56
HPT_WAIT_LIST_SIZE (114) DSANC 56
HPT_WAIT_LIST_START (108) DSANC 56
HPTYPE 226
HS_ARROW (2) DDCBC 37
HS_BLOCK_NAME (8) DDCBC 37
HS_DFH (3) DDCBC 37
HS_DOMID (6) DDCBC 37
HS_HASHTABLE (10) DDCBC 37
HS_LENGTH (0) DDCBC 37
HS_PREFIX (0) DDCBC 37
HS_READ_TOKEN (10) L2BL 209
HSANSAREA (0) L2HS 230
HSECB (0) L2HS 230
HSLENGTHBYTES (0) L2HS 230
HSMVSSTREAMTOKEN (0) L2HS 230
HSREADTOKEN 230
HSRETRSN (0) L2HS 230
HTB (0) PGHM 293
HTB_ABEND_TABLE (784) PGHM 293
HTB_AIDS_TABLE (5A4) PGHM 293
HTB_ARROW (2) PGHM 293
HTB_CONDITIONS_TABLE 293
HTB_DFH (3) PGHM 293
HTB_DOMID (6) PGHM 293
HTB_HTB (8) PGHM 293
HTB_LENGTH (0) PGHM 293
HTB_PREFIX (0) PGHM 293
HTB_PREV_TABLE 293
HTB_TABLES (18) PGHM 293
HTB_USED_RSAS (14) PGHM 293
HTE (0) PGHM 294
HTE_ABEND_PROGRAM (BIT) PGHM 294
HTE_ACTIVE (0) PGHM 294
HTE_COBOL_RSA (4) PGHM 294
HTE_DEFAULT (BIT) PGHM 294
HTE_EXECUTION_KEY (3) PGHM 294
HTE_IGNORE (BIT) PGHM 294
HTE_LABEL (4) PGHM 294
HTE_LABEL_AMODE_31 (BIT) PGHM 294
HTE_LABEL_BYTE (4) PGHM 294
HTE_LANGUAGE 294
HTE_PROGRAM (4) PGHM 294
HTE_PROGRAM_MASK (2) PGHM 294
HTE_USER_RSA (8) PGHM 294

I

ID (BIT) L2BL 208
ID_NOT_RECEIVED (BIT) CPCPS 33
ID_OR_NUMBER (0) L2LF 231, 233
ID_OR_NUMBER (10) L2LF 235

ID_OR_NUMBER (10) LGSF   200
ID_OR_NUMBER (14) L2LF   234
ID_OR_NUMBER (24) L2LF   235, 236
ID_OR_NUMBER (24) LGSF   200, 201
ID_OR_NUMBER (34) L2LF   235
ID_OR_NUMBER (34) LGSF   201
ID_OR_NUMBER (4) L2LF   234, 235
identity
    recovery manager identity instance, RMID   303
    recovery manager loggable object identity instance, RMLI   304
IDQ_DATATYPE (2C) FEP06   124
IDQ_INSTDISC (48) FEP06   124
IDQ_NAME_LENGTH (40) FEP06   124
IDQ_NUMBER (44) FEP06   124
IDQ_RECOVERY (4A) FEP06   124
IDQ_RES_NAME (30) FEP06   124
IDQ_RES_TYPE (49) FEP06   124
IDQDATA   124
IDT_COUNT (10) D2CSB   80
IDT_DISPOSITION (22) D2CSB   80
IDT_ENTRY (12) D2CSB   80
IDT_EYE   (2) D2CSB   80
IDT_LENGTH (0) D2CSB   80
IDT_PREFIX (0) D2CSB   80
IDT_URID (12) D2CSB   80
IFA   (0) SMDCC   352
IFA_END   (C) SMDCC   352
IFA_LENGTH (10) SMDCC   352
IFA_NEXT (0) SMDCC   352
IFA_PREV (4) SMDCC   352
IFA_START (8) SMDCC   352
IGNORE_SHUNT 1 NQPL   281
iliffe
    logger reusable extended iliffe vector class, RUEI   343
IN_COLD_STATE (54) RMSL   327, 329
IN_DEAD_TAIL (BIT) L2CH   221
IN_DISPATCHER_PRE_INIT   54
IN_STORE   (BIT) BAACT   16
IN_STORE_TARGET (0) BAACT   16
INBOUND_RECOVERY_IN_PROGRESS (BIT) RMLK   307, 310
INDEX   (114) RMUW   339
INDEX   (1C) L2CH   222
INDEX   (28C) L2BS   217
INDEX   (2A4) L2BS   218
INDEX   (474) RMLK   306
INDEX   (4C) L2CH   221
INDEX   (534) RMUW   339
INDEX   (54) RMLK   305
INDEX   (84) L2CH   221
INDEX   (C) L2RT   239
INDOUBT   (45) RMLS   319
INDOUBT   (A5) RMLK   313
INDOUBT   (A5) RMUW   332
INDOUBT   (FD) RMLK   314
INDOUBT   (FD) RMUW   333
INDOUBT_TIMEOUT_INTERVAL (54) RMLK   311
INDOUBT_TIMEOUT_INTERVAL (54) RMUW   331
info
    property set info, FEP13   136
INHERIT_SS (BIT) DSANC   57, 60
INIT_STATS_COLL (7A0) DMCB1   47
INIT_STATUS (14) CPSPS   34
INIT_STATUS (1C) PRS   296
INIT_SUSPEND_TOKEN   34, 296
INITIAL   (0) WRB   430
INITIAL_NO 1 WRB   430
INITIAL_YES 1 WRB   430
INITIALISED 1 DDCBC   37
INITIALISED 4 MEPS   259
INITIALISED 4 SMDCC   362
INITIALISED 4 TSA   381
INITIALISED 4 XMANC   437
INITIALISER (4C) RMDM   301
INITIALISER (50) L2DM   224
INITIALISING 4 SMDCC   362
INITIALISING 4 TSA   381
INITIALISING 4 XMANC   437
INITIATOR (16) RMLK   317
INITIATOR (6A) RMLK   310
INITIATOR (972) RMLK   308
INLINE_ACCESS_STRUCTURE   314, 333
inquire
inquire *(continued)*
    inquire application data xpi command, APIQ   2
INQUIRE_DISJOINT_CHAINS (2C) RMLI   304
INQUIRE_DISJOINT_CHAINS (8D4) RMLK   307
INQUIRE_DISJOINT_CHAINS (94) RMUW   338
INSERT_ELEMENT 1 MEMMS   256
INSERT1   1 MEMMS   256
INSERT10   1 MEMMS   256
INSERT2   1 MEMMS   256
INSERT3   1 MEMMS   256
INSERT4   1 MEMMS   256
INSERT5   1 MEMMS   256
INSERT6   1 MEMMS   256
INSERT7   1 MEMMS   256
INSERT8   1 MEMMS   256
INSERT9   1 MEMMS   256
instance
    recovery manager domain management instance, RMDM   301
    recovery manager identity instance, RMID   303
    recovery manager link instance, RMLK   309
    recovery manager link set instance, RMLS   318
    recovery manager loggable object identity instance, RMLI   304
    recovery manager logname instance, RMNM   321
    recovery manager logname set instance, RMNS   322
    recovery manager resource owner instance, RMRO   324
    recovery manager system log instance, RMSL   327
    recovery manager unit of work instance, RMUW   330
INSTANCE   (116) RMUW   339
INSTANCE   (476) RMLK   306
INSTANCE   (536) RMUW   339
INSTANCE   (56) RMLK   305
INSTANCE_COUNT (18) DSANC   58
INSTANCE_DATA_BLOCK   208, 215, 217, 218, 220, 221, 222, 229, 239, 247, 307, 309, 311, 313, 315, 318, 319, 324, 327, 329, 330, 332, 333, 334
INSTANCE_DATA_BLOCK (0) BAACT   5, 9, 21
INSTANCE_DATA_BLOCK (0) BAPT   23
INSTANCE_DATA_BLOCK (0) L2DM   224
INSTANCE_DATA_BLOCK (0) L2HP   226
INSTANCE_DATA_BLOCK (0) L2LT   238
INSTANCE_DATA_BLOCK (0) RMDM   301
INSTANCE_DATA_BLOCK (10) BAACT   17
INSTANCE_DATA_BLOCK (10) L2BL   209
INSTANCE_DATA_BLOCK (10) RMNM   321
INSTANCE_DATA_BLOCK (100) RMUW   339
INSTANCE_DATA_BLOCK (18) BAACT   10
INSTANCE_DATA_BLOCK (18) RMLI   304
INSTANCE_DATA_BLOCK (18) RMLK   316
INSTANCE_DATA_BLOCK (1C) RMDM   301
INSTANCE_DATA_BLOCK (20) L2DM   224
INSTANCE_DATA_BLOCK (38) L2BS   218
INSTANCE_DATA_BLOCK (38) L2CH   222
INSTANCE_DATA_BLOCK (38) L2SR   249
INSTANCE_DATA_BLOCK (38) RMNS   323
INSTANCE_DATA_BLOCK (4) BAACT   12, 18
INSTANCE_DATA_BLOCK (40) RMLK   305
INSTANCE_DATA_BLOCK (40) RMUW   337
INSTANCE_DATA_BLOCK (460) RMLK   306
INSTANCE_DATA_BLOCK (520) RMUW   339
INSTANCE_DATA_BLOCK (58) BAACT   6
INSTANCE_DATA_BLOCK (58) L2CH   221
INSTANCE_DATA_BLOCK (8) BAACT   6, 7
INSTANCE_DATA_BLOCK (80) RMUW   338
INSTANCE_DATA_BLOCK (88) RMNM   321
INSTANCE_DATA_BLOCK (880) RMLK   306
INSTANCE_DATA_BLOCK (8C0) RMLK   306
INSTANCE_DATA_BLOCK (90) L2CH   223
INSTANCE_DATA_BLOCK (A8) BAACT   18
INSTANCE_DATA_BLOCK (C8) BAACT   11
INSTANCE_LENGTH (12) BAACT   5
INSTANCE_LENGTH (18) BAACT   10
INSTANCE_VERSION (10) BAACT   5
INSTANCE_VERSION (1A) BAACT   10
INSTRUCTION_LENGTH (278) APLI   4
INT   (BIT) STUCB   376
interface
    external CICS interface control blocks, XCCBC   431
    frontend programming interface trace, FEP01   108
    frontend programming interface, FEP21   148
    language interface work area, APLI   3
    web business logic compatibility interface, WBA1C   413
    web business logic interface parameters, WBBLC   416
    web interface urp constants, WBUCC   424
INTERRUPT_CODE (27A) APLI   4

INTERRUPT_DATA 4
INTERVAL 1 DSTSK 67
INTERVAL_START 217, 249
INVALID_CLASS 1 SMMCC 366
INVALID_DATA (BIT) PAA 283
IO_IN_PROGRESS 4 L2BL 211
IO_IN_PROGRESS 4 L2HS 231
IS_ACT_LEN (4) BAACT 16
IS_ACT_LEN (E4) BAACT 16
IS_ACT_PTR (0) BAACT 16
IS_ACT_PTR (E0) BAACT 16
IS_PRO_LEN (C) BAACT 16
IS_PRO_LEN (EC) BAACT 16
IS_PRO_PTR (8) BAACT 16
IS_PRO_PTR (E8) BAACT 16
IS_TARGET (E0) BAACT 16
ITEMS (0) BAACT 9
ITEMS (58) BAACT 6
ITEMS (A8) BAACT 18
ITEMS (C8) BAACT 11
ITER0 6, 9, 11, 18, 213, 218, 222, 244, 249, 250, 305, 312, 313, 316, 318, 322, 323, 327, 329, 332, 337, 338, 339
ITERNODE (18) RMUW 335
ITERNODE (68) L2CH 223
IXG_STCK 216, 230, 248
IXGBRORD_COUNT (1D0) L2BS 216
IXGBRORD_COUNT (1D0) L2SR 248
IXGBRORD_COUNT (E0) L2HS 230
IXGBROST_COUNT (1CC) L2BS 216
IXGBROST_COUNT (1CC) L2SR 248
IXGBROST_COUNT (DC) L2HS 230
IXGDELET_COUNT (1D4) L2BS 216
IXGDELET_COUNT (1D4) L2SR 248
IXGDELET_COUNT (E4) L2HS 230
IXGWRITE_BYTES (1C0) L2BS 216
IXGWRITE_BYTES (1C0) L2SR 248
IXGWRITE_BYTES (D0) L2HS 230
IXGWRITE_COUNT (1BC) L2BS 216
IXGWRITE_COUNT (1BC) L2SR 248
IXGWRITE_COUNT (CC) L2HS 230
IXGWRITE_LATENCY (100) L2HS 230
IXGWRITE_LATENCY (1F0) L2BS 216
IXGWRITE_LATENCY (1F0) L2SR 248
IXGWRITE_STCK (1E8) L2BS 216
IXGWRITE_STCK (1E8) L2SR 248
IXGWRITE_STCK (F8) L2HS 230

J

JOURNAL_NAME (122) L2BS 215
JOURNAL_NAME (122) L2SR 247
JOURNAL_NAME (32) L2HS 230
JOURNAL_NAME (44) L2BL 208

K

KCB 151
KCB_ADD_CICS_RECOVERY_EP (40) KCB 151
KCB_ADD_DELTA (BIT) KCB 153
KCB_ALTERNATE_XRF_IDS (CC) KCB 153
KCB_ARROW (2) KCB 151
KCB_BLOCK_NAME (8) KCB 151
KCB_CANCEL_REQUESTED (BIT) KCB 152
KCB_CICS (BIT) KCB 152
KCB_CICS_SVC 153
KCB_CICS_SVC_NUMBER (F3) KCB 153
KCB_CLOCKING_ACTIVE 152
KCB_DATE_FORMAT 153
KCB_DDMMYY (BIT) KCB 153
KCB_DELTA_HIGH (F4) KCB 153
KCB_DELTA_LOW (F8) KCB 153
KCB_DESCRIPTION 153
KCB_DFH (3) KCB 151
KCB_DFHCRC_ADDRESS (110) KCB 154
KCB_DISPOSAL_CHAIN (128) KCB 154
KCB_DOMAIN_CALL (10) KCB 151
KCB_DOMAIN_NUMBER (88) KCB 152
KCB_DOMAIN_RETURN (18) KCB 151
KCB_DOMAIN_RETURN_24 (38) KCB 151
KCB_DOMAIN_TABLE (70) KCB 152
KCB_DOMAIN_TABLE_START (210) KCB 154
KCB_DOMAIN_VECTOR 154
KCB_DOMID (6) KCB 151
KCB_DUMP_REQUESTED (BIT) KCB 152
KCB_DUMP_RETRY (94) KCB 152
KCB_DYNAMIC_FIRST_FREE (120) KCB 154
KCB_DYNAMIC_GUARD (124) KCB 154
KCB_DYNAMIC_QUICK_CELL (120) KCB 154
KCB_ERROR_TABLE 152
KCB_ESTAE_ACTIVE (BIT) KCB 152
KCB_EXCESS_STATIC_TASKS (12C) KCB 154
KCB_FACILITY_STATUS 152
KCB_GATE_NUMBER (8C) KCB 152
KCB_GENERIC_APPLID (B4) KCB 153
KCB_GLOBAL_DATA_FLAGS (96) KCB 153
KCB_GMT_TO_LOCAL (FC) KCB 153
KCB_HPO_ACTIVE (BIT) KCB 152
KCB_IPL_STACK (E4) KCB 153
KCB_ISC_AVAILABLE (BIT) KCB 153
KCB_JOB_STEP_STATUS (54) KCB 152
KCB_KE_LOCK (158) KCB 154
KCB_KERNEL_STATUS (54) KCB 152
KCB_KTCB_NUMBER 153
KCB_KTCB_TABLE (7C) KCB 152
KCB_LENGTH (0) KCB 151
KCB_LOCAL_TIME_DELTA (F4) KCB 153
KCB_MASTER (BIT) KCB 153
KCB_MIN_FREE_OVERFLOWS 152
KCB_MMDDYY (BIT) KCB 153
KCB_MODULE_ADDRESS (8) KCB 154
KCB_MODULE_LENGTH (C) KCB 154
KCB_MODULE_VECTOR (0) KCB 154
KCB_MODULE_VECTOR_POINTER (E8) KCB 153
KCB_MXT_EXTRA_SEGMENTS_24 (114) KCB 154
KCB_MXT_EXTRA_SEGMENTS_31 (160) KCB 154
KCB_NORMAL_TERMINATION (BIT) KCB 152
KCB_NOTIFY_RESET_DOMAINS 153
KCB_NOTIFY_TRACE (BIT) KCB 153
KCB_OP_MODIFICATION (E3) KCB 153
KCB_OP_RELEASE (E2) KCB 153
KCB_OP_SYS (E0) KCB 153
KCB_OP_VERSION (E1) KCB 153
KCB_OUT_OF_STACK (BIT) KCB 152
KCB_OVERFLOW_STACK_LM_LOCK (4C) KCB 152
KCB_PARMS (A8) KCB 153
KCB_PARMS_ADDR (A8) KCB 153
KCB_PARMS_LEN (AC) KCB 153
KCB_PERCOLATE (14) KCB 151
KCB_PREFIX (0) KCB 151
KCB_PROCESS_OWN (0) KCB 151
KCB_QUIESCE_DOMAIN_RECEIVED (BIT) KCB 152
KCB_RECOVERY_EXIT (1C) KCB 151
KCB_RECOVERY_REQUEST (20) KCB 151
KCB_RESET_ADDRESS (24) KCB 151
KCB_RUNAWAY_LIMIT (48) KCB 152
KCB_SEG24_FIRST_FREE (60) KCB 152
KCB_SEG24_FREE_SEGS (66) KCB 152
KCB_SEG24_GUARD (64) KCB 152
KCB_SEG24_GUARD_COUNT (64) KCB 152
KCB_SEG24_QUICK_CELL 152
KCB_SEG31_FIRST_FREE (68) KCB 152
KCB_SEG31_FREE_SEGS (6E) KCB 152
KCB_SEG31_GUARD (6C) KCB 152
KCB_SEG31_GUARD_COUNT (6C) KCB 152
KCB_SEG31_QUICK_CELL (68) KCB 152
KCB_SET_DUB_ISSUED (BIT) KCB 153
KCB_SIT_NAME (D8) KCB 153
KCB_SPECIFIC_APPLID (BC) KCB 153
KCB_STATIC_FIRST_FREE (118) KCB 154
KCB_STATIC_GUARD (11C) KCB 154
KCB_STATIC_QUICK_CELL (118) KCB 154
KCB_STATIC_TASK_NUMBER (90) KCB 152
KCB_STIMER_ACTIVE (BIT) KCB 152
KCB_STIMER_INTERVAL (80) KCB 152
KCB_STK24_SUBPOOL_TOKEN (130) KCB 154
KCB_STK24E_SUBPOOL_TOKEN (140) KCB 154
KCB_STK31_SUBPOOL_TOKEN (138) KCB 154
KCB_STK31E_SUBPOOL_TOKEN (148) KCB 154
KCB_STORAGE_PROTECT_SUPPORTED (BIT) KCB 153
KCB_SUBROUTINE_CALL (28) KCB 151
KCB_SUBROUTINE_RETURN (2C) KCB 151
KCB_SUBROUTINE_RETURN_24 (3C) KCB 151
KCB_SUBTRACT_DELTA (BIT) KCB 153
KCB_SYSID (D4) KCB 153
KCB_SYSTEM_MASTER (BIT) KCB 153

KCB_TASK_CHAIN_START (58) KCB 152
KCB_TASK_SUBPOOL_TOKEN (150) KCB 154
KCB_TEMP_STATIC_TASK_NUMBER (44) KCB 151
KCB_TERMINATE_REQUESTED (BIT) KCB 152
KCB_TIMER_ACTIVE (BIT) KCB 153
KCB_TIMER_CHANGES 153
KCB_TIMER_STATE (A4) KCB 153
KCB_TIMER_STATUS (56) KCB 152
KCB_TRACE (100) KCB 153
KCB_TRACE_COUNT 153
KCB_TRACE_DOM_CALL (30) KCB 151
KCB_TRACE_DOM_TABLE (34) KCB 151
KCB_TRAP (104) KCB 153
KCB_TRAP_ACTIVE 152
KCB_TRAP_ADDRESS (108) KCB 153
KCB_TRAP_ENABLED (BIT) KCB 153
KCB_TRAP_PARAMETER (10C) KCB 153
KCB_TRAP_STATUS (104) KCB 153
KCB_TRMF (100) KCB 153
KCB_VECTOR_ENTRY 154
KCB_VECTOR_SIZE (0) KCB 154
KCB_WINDOW_VECTOR_POINTER (EC) KCB 153
KCB_XRF (BIT) KCB 153
KCB_XRF_COMMAND_LIST (C4) KCB 153
KCB_YYMMDD (BIT) KCB 153
KE_TASK_TOKEN (24) DSANC 58
KECB 155
KEMHD 161
KERN_ANCHOR (178) DSANC 56
KERN_DTE (0) KESTP 163
KERN_DTE_ANCHOR 164
KERN_DTE_INDEX 163
KERN0DCL 1 KESTP 164
KERN0KER 1 KESTP 164
KERN0LCL 1 KESTP 164
KERN0SCL 1 KESTP 164
KERNACR (BIT) KESTP 163
KERNBPTR (4) KESTP 163
KERNDFAB (BIT) KESTP 163
KERNDTAB (58) KESTP 163
kernel
  kernel anchor block, KCB 151
  kernel control blocks, KECB 155
  kernel module header, KEMHD 161
  kernel stack entry, KESTP 163
KERNEL_TASKID (64) DSTSK 65
KERNERRD (BIT) KESTP 163
KERNLCON (BIT) KESTP 163
KERNLOOP (BIT) KESTP 163
KERNMODH (64) KESTP 163
KERNMODS 163
KERNNAB (60) KESTP 163
KERNOFF0 163
KERNOFLN (2) KESTP 163
KERNPOWN (54) KESTP 163
KERNREGS (C) KESTP 163
KERNRGST 163
KERNSAVE (BIT) KESTP 163
KERNSAVP (4C) KESTP 163
KERNSCCN (70) KESTP 163
KERNSGCN (68) KESTP 163
KERNSTAT (1) KESTP 163
KERNSTCK (0) KESTP 163
KERNSTCK_END 163
KERNTASN (50) KESTP 163
KERNTRFL (5C) KESTP 163
KERR_PTR (3C) TIA 378
KES_AUTOMATIC (C4) KESTP 163
KES_HEADER (0) KESTP 163
KES_LENGTH (C0) KESTP 163
KES_REGISTERS (80) KESTP 163
KES_SAVED_STACK_ENTRY (0) KESTP 163
KESTACKSAVE (0) KESTP 163
KESTP 163
KEY (10) BAACT 6, 7
KEY (C) BAACT 12, 18
KEY_LENGTH 2 CCGD 31
KEYPOINT_CHAIN (1F) RMSL 327, 329
KEYPOINT_COUNT (2E) RMLK 311
KEYPOINT_COUNT (2E) RMUW 330
KEYPOINT_MOVE_LOG_RECORD 4 RMUW 336, 340
KEYPOINT_SCHEDULED (1C) RMSL 327, 329
KEYPOINTED_FOR_MOVE (BIT) RMLK 312
KEYPOINTED_FOR_MOVE (BIT) RMUW 331
KNOWN_BY (28) L2BL 208
KNOWN_INSTANCES (18) RMNS 323
KTCB_ABEND_999 (3B) KECB 160
KTCB_ACCUM_TIME (18) KECB 159
KTCB_ACTIVE_TASK (10) KECB 159
KTCB_ARBITRARY_NAME 1 KECB 161
KTCB_ATTACH_INIT_ECB (4C) KECB 160
KTCB_ATTACH_INTERFACE (48) KECB 160
KTCB_ATTACH_PARAM (48) KECB 160
KTCB_ATTACH_TCB_ADDRESS (50) KECB 160
KTCB_ATTACHED_TCB (BIT) KECB 159
KTCB_ATTACHING_TCB (BIT) KECB 160
KTCB_AUTOMATIC_END (FE0) KECB 161
KTCB_CANCEL_ESTAE (BIT) KECB 160
KTCB_CANCEL_REQUESTED (BIT) KECB 160
KTCB_CANCEL_STATE (92) KECB 160
KTCB_CLEAN_UP_ESTAE 160
KTCB_CONCURRENT 1 KECB 161
KTCB_CURRENTLY_ATTACHED (BIT) KECB 159
KTCB_DAUGHTER_TERMINATED (BIT) KECB 160
KTCB_DEFAULT_TASK (C) KECB 159
KTCB_ENTRY (0) KECB 159
KTCB_ERROR_MAX_EXCEEDED (BIT) KECB 160
KTCB_ESSENTIAL_TCB (39) KECB 159
KTCB_ESTAE_AUTOMATIC (A0) KECB 161
KTCB_ESTAE_ENVIRONMENT (BIT) KECB 160
KTCB_ESTAE_STATE 160
KTCB_ETXR_AUTOMATIC (C80) KECB 161
KTCB_EXEC_CAPABLE (BIT) KECB 159
KTCB_EXIT_TIME (28) KECB 159
KTCB_FILE_OWNING 1 KECB 161
KTCB_HAS_BEEN_DETACHED (BIT) KECB 160
KTCB_HEADER (0) KECB 159
KTCB_JOB_STEP 1 KECB 161
KTCB_KESTX_IN_PROGRESS (BIT) KECB 160
KTCB_KETIX_LAST_INVOKED 160
KTCB_LE_CICS (BIT) KECB 159
KTCB_LOCK_ACTIVE_QEL_PTR (6C) KECB 160
KTCB_LOCK_BACK_POINTER (68) KECB 160
KTCB_LOCK_CHAIN 160
KTCB_LOCK_ECB (70) KECB 160
KTCB_LOCK_ELEMENT (60) KECB 160
KTCB_LOCK_LCB_PTR (68) KECB 160
KTCB_LOCK_STATIC_QEL (60) KECB 160
KTCB_MODENAME (3D) KECB 160
KTCB_MOTHER_KTCB (8C) KECB 160
KTCB_MVS_RSA (58) KECB 160
KTCB_NAME (0) KECB 159
KTCB_NEXT_ENTRY (88) KECB 160
KTCB_NEXT_FREE (8) KECB 159
KTCB_NO_SDWA (BIT) KECB 160
KTCB_ONC_RPC 1 KECB 161
KTCB_OUT_OF_STACK (BIT) KECB 160
KTCB_PERCOLATE_ERROR (BIT) KECB 160
KTCB_PRTY_RELATIVE_TO_PARENT (90) KECB 160
KTCB_QUASI_REENTRANT 1 KECB 161
KTCB_RESET_FP_REGS (84) KECB 160
KTCB_RESET_PARAMETER (5C) KECB 160
KTCB_RESET_REQUESTED (BIT) KECB 160
KTCB_RESOURCE_OWNING 1 KECB 161
KTCB_RUNAWAY_REQUESTED (BIT) KECB 160
KTCB_SECONDARY_LU 1 KECB 161
KTCB_SS_ENV (BIT) KECB 159
KTCB_STATE (38) KECB 159
KTCB_STEAL_POINT (14) KECB 159
KTCB_STIMER_AUTOMATIC (B30) KECB 161
KTCB_STIMER_TIME (20) KECB 159
KTCB_SWITCH_SS_ENV (BIT) KECB 159
KTCB_TCB_AUTOMATIC (CC0) KECB 161
KTCB_TCB_POSTED (BIT) KECB 159
KTCB_TCB_TOKEN (74) KECB 160
KTCB_TCB_TYPE 160
KTCB_TCB_WAIT_ECB (34) KECB 159
KTCB_TERMINATE_ECB (54) KECB 160
KTCB_TIMER (18) KECB 159
KTCB_TIMER_ACTIVE (BIT) KECB 159
KTCB_TIMER_CHANGES 159
KTCB_TIMER_STATE (30) KECB 159
KTCB_TRAP_PARAMETER (40) KECB 160
KTCB_UNUSED (BIT) KECB 159
KTCH_ARROW (2) KECB 159
KTCH_BLOCK_NAME (8) KECB 159

KTCH_DFH (3) KECB 159
KTCH_DOMID (6) KECB 159
KTCH_ENTRY_LENGTH (18) KECB 159
KTCH_FIRST_FREE (30) KECB 159
KTCH_FO_TCB (24) KECB 159
KTCH_GUARD (34) KECB 159
KTCH_LAST_ENTRY (14) KECB 159
KTCH_LENGTH (0) KECB 159
KTCH_PREFIX (0) KECB 159
KTCH_QR_TCB (2C) KECB 159
KTCH_QUICK_CELL (30) KECB 159
KTCH_RO_TCB (28) KECB 159
KTCH_SPECIFIC_TCBS (20) KECB 159
KTCH_STEP_TCB (20) KECB 159
KTCH_TABLE_START (10) KECB 159

L

L2_EYE_LEN (0) L2BL 209
L2_EYE_LEN (0) L2BS 218
L2_EYE_LEN (0) L2CH 222
L2_EYE_LEN (0) L2DM 224
L2_EYE_LEN (0) L2SL 240
L2_EYE_LEN (0) L2SR 249
L2_EYE_LEN (10) L2BL 209
L2_EYE_LEN (260) L2BS 217
L2_EYE_LEN (38) L2BS 219
L2_EYE_LEN (38) L2CH 222
L2_EYE_LEN (38) L2SR 249
L2_EYE_LEN (8) L2BL 208
L2_EYE_LEN (8) L2BS 212
L2_EYE_LEN (8) L2CH 220
L2_EYE_LEN (8) L2HS 229
L2_EYE_LEN (8) L2SR 243
L2_EYE_LEN (F8) L2BS 215
L2_EYE_LEN (F8) L2SR 247
L2_EYE_OFFSET (12) L2BL 209
L2_EYE_OFFSET (2) L2BL 209
L2_EYE_OFFSET (2) L2BS 218
L2_EYE_OFFSET (2) L2CH 222
L2_EYE_OFFSET (2) L2DM 224
L2_EYE_OFFSET (2) L2SL 240
L2_EYE_OFFSET (2) L2SR 249
L2_EYE_OFFSET (262) L2BS 217
L2_EYE_OFFSET (3A) L2BS 219
L2_EYE_OFFSET (3A) L2CH 222
L2_EYE_OFFSET (3A) L2SR 249
L2_EYE_OFFSET (A) L2BL 208
L2_EYE_OFFSET (A) L2BS 212
L2_EYE_OFFSET (A) L2CH 220
L2_EYE_OFFSET (A) L2HS 229
L2_EYE_OFFSET (A) L2SR 243
L2_EYE_OFFSET (FA) L2BS 215
L2_EYE_OFFSET (FA) L2SR 247
L2_EYE_STRING (14) L2BL 209
L2_EYE_STRING (264) L2BS 217
L2_EYE_STRING (3C) L2BS 219
L2_EYE_STRING (3C) L2CH 222
L2_EYE_STRING (3C) L2SR 249
L2_EYE_STRING (4) L2BL 209
L2_EYE_STRING (4) L2BS 218
L2_EYE_STRING (4) L2CH 222
L2_EYE_STRING (4) L2DM 224
L2_EYE_STRING (4) L2SL 241
L2_EYE_STRING (4) L2SR 249
L2_EYE_STRING (C) L2BL 208
L2_EYE_STRING (C) L2BS 212
L2_EYE_STRING (C) L2CH 220
L2_EYE_STRING (C) L2HS 229
L2_EYE_STRING (C) L2SR 243
L2_EYE_STRING (FC) L2BS 215
L2_EYE_STRING (FC) L2SR 247
L2BL 208
L2BL_CLASSID 4 L2DM 225
L2BS 211
L2BS_CLASSID 4 L2DM 225
L2CH 219
L2CH_CLASSID 4 L2DM 225
L2CH_LOCK_ERROR_CODE 4 L2CH 224
L2CH_UNLOCK_ERROR_CODE 4 L2CH 224
L2CH_WRONG_TCB_ERROR_CODE 4 L2CH 224
L2DM 224
L2DM *(continued)*
  log manager l2dm class, L2DM 224
L2DM (0) L2DM 224
L2DM_CLASS_MANAGER (20) L2DM 224
L2DM_EYE_CATCHER (0) L2DM 224
L2DM_INITIALISED 4 L2DM 225
L2DM_INITIALISING 4 L2DM 225
L2DM_LOCK_ERROR_CODE 4 L2DM 225
L2DM_LOCK_FREE 4 L2DM 225
L2DM_LOCK_HELD 4 L2DM 225
L2DM_LOCK_STATUS 225
L2DM_LOCK_TOKEN (1C) L2DM 224
L2DM_NUM_CLASSES 4 L2DM 225
L2DM_PNAME 16 L2DM 225
L2DM_PTYPE 8 L2DM 225
L2DM_QUIESCED 4 L2DM 225
L2DM_QUIESCING 4 L2DM 225
L2DM_STATE (10) L2DM 224
L2DM_SUBPOOL 224
L2DM_TERMINATED 4 L2DM 225
L2DM_TERMINATING 4 L2DM 225
L2DM_UNLOCK_ERROR_CODE 4 L2DM 225
L2HP 226
L2HS 227
L2LF 231
L2LT 238
L2RT 239
L2SL 240
L2SL_CLASSID 4 L2DM 225
L2SR 242
L2SR_CLASSID 4 L2DM 225
L2SR_LOCK_ERROR_CODE 4 L2SR 251
L2SR_UNLOCK_ERROR_CODE 4 L2SR 251
L2VP_CLASSID 4 L2DM 225
LAFPB (0) LDCBS 171
LAFPB_ABEND 171
LAFPB_ARROW (2) LDCBS 171
LAFPB_BAD_CONCATNO 1 LDCBS 176
LAFPB_BAD_PARM 1 LDCBS 176
LAFPB_BAD_STORAGE 1 LDCBS 176
LAFPB_BLDL_PLIST (1C) LDCBS 171
LAFPB_BLOCK_ID (8) LDCBS 171
LAFPB_CALR 1 LDCBS 176
LAFPB_CLOSE_ERROR 1 LDCBS 176
LAFPB_CREATION_STCK (24) LDCBS 171
LAFPB_DESERV_AREA (2C) LDCBS 171
LAFPB_DESERV_AREAL (30) LDCBS 171
LAFPB_DFH (3) LDCBS 171
LAFPB_DOMAIN (6) LDCBS 171
LAFPB_ENVR 1 LDCBS 176
LAFPB_EXTENT_ERROR 1 LDCBS 176
LAFPB_FUNCTION (10) LDCBS 171
LAFPB_ID_STRING 8 LDCBS 176
LAFPB_INFO 1 LDCBS 176
LAFPB_INVALID_FUNCTION 1 LDCBS 176
LAFPB_IOERR 1 LDCBS 176
LAFPB_IS_PDS 1 LDCBS 176
LAFPB_LENGTH (0) LDCBS 171
LAFPB_LOAD_POINT (20) LDCBS 171
LAFPB_NO_AUTHORISATION 1 LDCBS 176
LAFPB_NO_DD 1 LDCBS 176
LAFPB_NO_FESTAE 1 LDCBS 176
LAFPB_NOSTORE 1 LDCBS 176
LAFPB_NOT_CONNECTED 1 LDCBS 176
LAFPB_NOT_EXECUTABLE 1 LDCBS 176
LAFPB_NOTFOUND 1 LDCBS 176
LAFPB_OK 1 LDCBS 176
LAFPB_OPEN_ERROR 1 LDCBS 176
LAFPB_PARM 1 LDCBS 176
LAFPB_PREFIX (0) LDCBS 171
LAFPB_R0 (18) LDCBS 171
LAFPB_REASON (16) LDCBS 171
LAFPB_RESPONSE (11) LDCBS 171
LAFPB_RPL_BLDL 1 LDCBS 176
LAFPB_RPL_CLOSE 1 LDCBS 176
LAFPB_RPL_DISCONNECT 1 LDCBS 176
LAFPB_RPL_END 1 LDCBS 176
LAFPB_RPL_GET_SMDE 1 LDCBS 176
LAFPB_RPL_LLACOPY 1 LDCBS 176
LAFPB_RPL_LOAD 1 LDCBS 176
LAFPB_RPL_LOAD_WITH_PMAR 1 LDCBS 176
LAFPB_RPL_OPEN 1 LDCBS 176
LAFPB_UNKNOWN_ERROR 1 LDCBS 176

LAFPB_WARN 1 LDCBS 176
LANG_ENV_REASON_CODE (20) APLI 3
LANG_ENV_RSA (114) APLI 3
LANG_ENV_WORKAREA (24) APLI 3
language
  language interface work area, APLI 3
LANGUAGE_BITS (2A8) APLI 5
LANGUAGE_INTERFACE_WORKAREA (0) APLI 3
LANGUAGES_USED (28) MEPS 257
LAST 308, 310
LAST (14) DDBSC 35
LAST_BLOCK_ID (8) L2BL 209
LAST_BLOCK_ID (D8) L2BS 213
LAST_BLOCK_ID (D8) L2SR 244
LAST_BLOCK_TIME (10) L2BL 209
LAST_BLOCK_TIME (E0) L2BS 213
LAST_BLOCK_TIME (E0) L2SR 244
LAST_CICS_CMD_REGISTERS_ADDR (28C) APLI 4
LAST_FORCE_TASK (258) L2BS 217
LAST_FORCE_TASK (258) L2SR 249
LAST_ID 57, 60
LAST_RESET_TIME 57
LAST_SMF_RC (3F) STCB1 374
LBH 395
LBH_M (8) TSOL 395
LBH_N (4) TSOL 395
LBH_P (0) TSOL 395
LD_APE_CELL_POOL (20) LDCBS 170
LD_CICS_COLD_STARTED (BIT) LDCBS 169
LD_CICS_INITIALISED (BIT) LDCBS 169
LD_CONTROL_POOL (18) LDCBS 169
LD_CPE_CELL_POOL (30) LDCBS 170
LD_CSECTL_CELL_POOL (28) LDCBS 170
LD_DC_EPADDR (2A4) LDCBS 171
LD_DFHSIP_EPADDR (2A8) LDCBS 171
LD_DOMAIN_STATUS (10) LDCBS 169
LD_DSA_NIU_Q_SIZE (1F8) LDCBS 171
LD_DSA_NIU_Q_TIME (1F0) LDCBS 171
LD_DSA_PROG_REMOVALS (1E8) LDCBS 170
LD_DSA_RECLAIMS (1EC) LDCBS 170
LD_DSA_RECORDS (1DC) LDCBS 170
LD_DSA_RPS (1E0) LDCBS 170
LD_DSA_TARGET (1E4) LDCBS 170
LD_DSA_USAGE (1DC) LDCBS 170
LD_DUMMY_CDE_POOL (38) LDCBS 170
LD_FLAGS (14) LDCBS 169
LD_GLOBAL_CATALOG_IN_USE (BIT) LDCBS 169
LD_LIBRARY_LOCK (D4) LDCBS 170
LD_LLACOPY_IN_REFRESH (BIT) LDCBS 169
LD_LLACOPY_NEWCOPY 1 LDCBS 175
LD_LLACOPY_NO 1 LDCBS 175
LD_LLACOPY_STATUS (16) LDCBS 169
LD_LLACOPY_YES 1 LDCBS 175
LD_LONG_NAME_CACHE_INVALID 8 LDCBS 176
LD_LONG_NAME_CACHE_KEYL 4 LDCBS 176
LD_LONG_NAME_CACHE_NAME 4 LDCBS 176
LD_LONG_NAME_CACHE_TOKEN (38C) LDCBS 171
LD_LONG_NAME_NOT_CACHED 8 LDCBS 176
LD_LONG_NAME_NOT_IN_RPL 8 LDCBS 176
LD_LPA_IN_USE 1 LDCBS 175
LD_LPA_NOT_IN_USE 1 LDCBS 175
LD_LPA_STATUS (13) LDCBS 169
LD_NT_EPADDR (29C) LDCBS 171
LD_RPL_CLOSED 1 LDCBS 175
LD_RPL_OPEN 1 LDCBS 175
LD_RPL_STATUS (12) LDCBS 169
LD_SLD (17) LDCBS 169
LD_ST_EPADDR (2A0) LDCBS 171
LD_STATE_LOCK (D0) LDCBS 170
LD_STATS_BUFFER_PTR (180) LDCBS 170
LD_STATS_BUFFER_SIZE 4 LDCBS 176
LD_STORAGE_FACTOR (1D8) LDCBS 170
LD_SUBPOOL_DATA (2BC) LDCBS 171
LD_SUBPOOL_DATA2 (40) LDCBS 170
LD_XLDELETE_ACTIVE (BIT) LDCBS 169
LD_XLDLOAD_ACTIVE (BIT) LDCBS 169
LDBE (0) LDCBS 172
LDBE_ANCHOR (14C) LDCBS 170
LDBE_ANCHOR_ID 8 LDCBS 177
LDBE_ARROW (2) LDCBS 172
LDBE_BLOCK_ID (8) LDCBS 172
LDBE_CHAIN_SIZE (148) LDCBS 170
LDBE_CREATION_STCK (2C) LDCBS 172
LDBE_DFH (3) LDCBS 172
LDBE_DOMAIN (6) LDCBS 172
LDBE_ID_STRING 8 LDCBS 176
LDBE_LAST_APE_ADDRESS (1C) LDCBS 172
LDBE_LAST_CPE_ADDRESS (18) LDCBS 172
LDBE_LAST_ENTRY_POINT (20) LDCBS 172
LDBE_LAST_PROGRAM_NAME (24) LDCBS 172
LDBE_LENGTH (0) LDCBS 172
LDBE_NEXT (10) LDCBS 172
LDBE_PREFIX (0) LDCBS 172
LDBE_PRIOR (14) LDCBS 172
LDCBS 164
LDDU_ABEND 8 LDCBS 174
LDDU_BAD_LOB 8 LDCBS 174
LDDU_BAD_PDB 8 LDCBS 174
LDDU_BAD_STRUCTURE 8 LDCBS 174
LDDU_LOOP 8 LDCBS 174
LDDU_SEVERE_ERROR 8 LDCBS 174
LDMATCH_ERROR_CODE 4 LGANC 193
LDME_ABEND 4 LDCBS 177
LDME_ADD_GATE_FAILED 4 LDCBS 177
LDME_BAD_OPEN 4 LDCBS 177
LDME_BAD_PDB 4 LDCBS 177
LDME_BLDL_LIMIT_EXCEEDED 4 LDCBS 177
LDME_CC_LOB_BAD 4 LDCBS 177
LDME_CONBLOK_INVALID 4 LDCBS 177
LDME_LIBRARY_IO_ERROR 4 LDCBS 177
LDME_LOOP 4 LDCBS 177
LDME_NO_MODULE 4 LDCBS 177
LDME_NO_NT_MODULE 4 LDCBS 177
LDME_NO_OS_STORAGE 4 LDCBS 177
LDME_NO_ST_MODULE 4 LDCBS 177
LDME_NOT_IN_LPA 4 LDCBS 177
LDME_SEVERE_ERROR 4 LDCBS 177
LDWE (0) LDCBS 172
LDWE_ANCHOR (130) LDCBS 170
LDWE_ANCHOR_ID 8 LDCBS 177
LDWE_ARROW (2) LDCBS 172
LDWE_BLOCK_ID (8) LDCBS 172
LDWE_CHAIN_SIZE (12C) LDCBS 170
LDWE_CPE_ADDRESS (1C) LDCBS 172
LDWE_CREATION_STCK (28) LDCBS 172
LDWE_DFH (3) LDCBS 172
LDWE_DOMAIN (6) LDCBS 172
LDWE_ID_STRING 8 LDCBS 177
LDWE_LENGTH (0) LDCBS 172
LDWE_NEXT (10) LDCBS 172
LDWE_PREFIX (0) LDCBS 172
LDWE_PRIOR (14) LDCBS 172
LDWE_PROGRAM_NAME (20) LDCBS 172
LDWE_RESUME_NO 4 LDCBS 177
LDWE_RESUME_REQUIRED (30) LDCBS 172
LDWE_RESUME_YES 4 LDCBS 177
LDWE_SUSPEND_TOKEN (18) LDCBS 172
LE_CICS (BIT) DSANC 57, 60
LE_COMP_AND_SWAP_SECTION (C) LMCB1 205
LE_CS_SUSPEND 205
LE_DELETED 205
LE_MODE_S 205
LE_NEXT_PTR (4) LMCB1 205
LE_OWNER (0) LMCB1 205
LE_PURGED (BIT) LMCB1 205
LE_STATUS (10) LMCB1 205
LE_SUSPEND_TOKEN (8) LMCB1 205
LE370_THREAD_TOKEN (8) APLI 3
LE370_THREAD_WORKAREA_ADDR (18) APLI 3
LEFT (0) DDBSC 35
LEN (0) DSTBA 63
LEN (17) USANC 405
LEN (1F) UDB 403
LEN (2B) UDB 403
LEN (3) XSSS 454
LEN (37) UDB 404
LEN (3C) L2BL 208
LEN (4) L2BL 209
LEN (67) XSSS 452
LEN (77) XSSS 452
LEN (87) XSSS 452
LENGTH_DATA_WRITTEN (6C) STCB1 375
LENGTH_DFHEIBLK 4 PGA 286
LENGTH_EISTACKA 4 PGA 286
LENGTH_EISUPERB 4 PGA 286
LENGTH_EIUS_STACK_AREA 4 PGA 286

LENGTH_EIUS_SUPER_STACK 4 PGA 286
LENGTH_TCAPCTWA 4 PGA 286
LEVEL (18) BAPT 23
LEVEL (20) SOA 372
LEVEL1 (BIT) XCCBC 431
LEVEL2 (BIT) XCCBC 431
LF_PLIST_DID (2) KEMHD 162
LF_PLIST_DLN (4) KEMHD 162
LF_PLIST_LEN (0) KEMHD 162
LF_PLIST_MDC (E) KEMHD 162
LF_PLIST_MOD (C) KEMHD 162
LF_PLIST_MODULE_OFFSET (6) KEMHD 162
LF_PLIST_TRC (8) KEMHD 162
LF_PLIST_TRCN 162
LF_PLIST_TRF (10) KEMHD 162
LF_PLIST_TRIC (BIT) KEMHD 162
LF_PLIST_TRRN (BIT) KEMHD 162
LF_PLIST_TRTR (BIT) KEMHD 162
LG_LGUOW_LOCK_NAME 8 LGANC 193
LG_LOCK_NAME 8 LGANC 193
LG_LOGOFLOG 8 LGANC 193
LG_STATE_INITIALISED 1 LGANC 192
LG_STATE_INITIALISING 1 LGANC 192
LG_STATE_QUIESCED 1 LGANC 192
LG_STATE_QUIESCING 1 LGANC 192
LG_STATE_TERMINATED 1 LGANC 192
LG_STATS_BUFFER_PTR (64) LGANC 189
LG_STATS_BUFFER_SIZE 4 LGANC 192
LG_STREAM_LOCK_NAME 8 LGANC 193
LGA (0) LGANC 188
LGA_APPLID (8D) LGANC 189
LGA_APPLID_L (8D) LGANC 189
LGA_APPLID_N (8E) LGANC 189
LGA_BLOCKNAME 8 LGANC 193
LGA_BR_HDR_PTR (60) LGANC 188
LGA_BR_SUBPOOL_TOKEN (40) LGANC 188
LGA_COLD_START (BIT) LGANC 188
LGA_END 189
LGA_EYE_CATCHER 14 LGANC 193
LGA_FLAGS (15) LGANC 188
LGA_GD_HDR_PTR (54) LGANC 188
LGA_GD_SUBPOOL_TOKEN (28) LGANC 188
LGA_GENERAL_SPTOKEN 188
LGA_INITIAL_START (BIT) LGANC 188
LGA_JI_HDR_PTR (58) LGANC 188
LGA_JI_SUBPOOL_TOKEN (30) LGANC 188
LGA_JM_HDR_PTR (5C) LGANC 188
LGA_JM_SUBPOOL_TOKEN (38) LGANC 188
LGA_JN_ENQPOOL_TOKEN (78) LGANC 189
LGA_L2_ACTIVE (BIT) LGANC 189
LGA_L2_FLAGS (96) LGANC 189
LGA_L2_PART 189
LGA_LAST_JNL_RESET_TIME (68) LGANC 189
LGA_LAST_LSN_RESET_TIME (70) LGANC 189
LGA_LENGTH (0) LGANC 188
LGA_LG_PART (0) LGANC 188
LGA_LG_STATE (14) LGANC 188
LGA_LGUOW_LOCK_TOKEN 189
LGA_LOCK_TOKEN (10) LGANC 188
LGA_PREFIX (0) LGANC 188
LGA_PREFIX_TEXT (2) LGANC 188
LGA_SD_HDR_PTR (50) LGANC 188
LGA_SD_SUBPOOL_TOKEN (20) LGANC 188
LGA_SMF_LOCK_TOKEN (80) LGANC 189
LGA_ST_ENQPOOL_TOKEN (7C) LGANC 189
LGA_STATSBUFFER 8 LGANC 193
LGA_SYSID (9C) LGANC 189
LGA_SYSID_L (9C) LGANC 189
LGA_SYSID_N (9D) LGANC 189
LGA_USERID (84) LGANC 189
LGA_USERID_L (84) LGANC 189
LGA_USERID_N (85) LGANC 189
LGA_UW_SUBPOOL_TOKEN (48) LGANC 188
LGA_XLGSTRM_ACTIVE (BIT) LGANC 188
LGA_XLGWBC_ACTIVE (BIT) LGANC 188
LGA_XRSINDI_ACTIVE (BIT) LGANC 188
LGANC 188
LGBH_BLOCK_INFO (20) L2LF 232
LGBH_BLOCK_INFO (20) LGSF 199
LGBH_BLOCK_INFO (58) L2BL 210
LGBH_BLOCK_NUMBER (20) L2LF 232
LGBH_BLOCK_NUMBER (20) LGSF 199
LGBH_BLOCK_NUMBER (58) L2BL 210
LGBH_BLOCK_TYPE (0) L2LF 231, 232
LGBH_BLOCK_TYPE (0) LGSF 199
LGBH_BLOCK_TYPE (38) L2BL 210
LGBH_BLOCK_TYPE_ARROW 1 L2LF 237
LGBH_BLOCK_TYPE_DFH 3 L2LF 237
LGBH_BLOCK_VER (3E) L2BL 210
LGBH_BLOCK_VER (6) L2LF 231, 232
LGBH_BLOCK_VER (6) LGSF 199
LGBH_BLOCK_VERSION_NO 2 L2LF 237
LGBH_BT_ARROW (0) L2LF 231, 232
LGBH_BT_ARROW (0) LGSF 199
LGBH_BT_ARROW (38) L2BL 210
LGBH_BT_DFH (1) L2LF 231, 232
LGBH_BT_DFH (1) LGSF 199
LGBH_BT_DFH (39) L2BL 210
LGBH_CICS_INFO (40) L2BL 210
LGBH_CICS_INFO (8) L2LF 231, 232
LGBH_CICS_INFO (8) LGSF 199
LGBH_DATA (28) L2LF 232
LGBH_DATA (28) LGSF 199
LGBH_DATA (60) L2BL 210
LGBH_FLAGS (3D) L2BL 210
LGBH_FLAGS (5) L2LF 231, 232
LGBH_FLAGS (5) LGSF 199
LGBH_GENERIC_APPLID (40) L2BL 210
LGBH_GENERIC_APPLID (8) L2LF 231, 232
LGBH_GENERIC_APPLID (8) LGSF 199
LGBH_GLOBAL_INFO (0) L2LF 231, 232
LGBH_GLOBAL_INFO (0) LGSF 199
LGBH_GLOBAL_INFO (38) L2BL 210
LGBH_LOG_TYPE 199, 210, 231, 232
LGBH_LOG_TYPE_GENERAL 1 L2LF 237
LGBH_LOG_TYPE_SYSTEM 1 L2LF 237
LGBH_START_GMT (10) L2LF 232
LGBH_START_GMT (10) LGSF 199
LGBH_START_GMT (48) L2BL 210
LGBH_START_LOCAL (18) L2LF 232
LGBH_START_LOCAL (18) LGSF 199
LGBH_START_LOCAL (50) L2BL 210
LGBR_BLOCKING 4 LGANC 192
LGBR_BROWSE_DATA (0) LGANC 191
LGBR_BROWSE_TOKEN (0) LGANC 191
LGBR_JMNAME (5) LGANC 191
LGBR_JNAME (5) LGANC 191
LGBR_KEY (5) LGANC 191
LGBR_STREAM (5) LGANC 191
LGBR_TYPE (4) LGANC 191
LGFL 198
LGFL_DATA_TYPE (0) LGFL 198
LGFL_JNL_FAIL_REC 2 LGFL 198
LGFL_JNL_NAME (1C) LGFL 198
LGFL_RECORD (0) LGFL 198
LGFL_STREAM_FAIL_REC 2 LGFL 198
LGFL_STREAM_NAME (2) LGFL 198
LGGD_BLOCKING 4 LGANC 192
LGGD_COMPONENT (18) LGANC 190
LGGD_DOMAIN_NO 190
LGGD_ERROR_GATE (20) LGANC 190
LGGD_GLOG_DATA (0) LGANC 190
LGGD_JNAME (10) LGANC 190
LGGD_LOG_TOKEN (0) LGANC 190
LGGD_LOGTYPE (1A) LGANC 190
LGGD_STREAM_TOKEN (C) LGANC 190
LGGD_USER_TOKEN (4) LGANC 190
LGJI_FAIL_REASON (25) LGANC 190
LGJI_JNAME (0) LGANC 190
LGJI_JNLFLUSH_REQS (38) LGANC 190
LGJI_JNLWRITE_BYTES (30) LGANC 190
LGJI_JNLWRITE_COUNT (2C) LGANC 190
LGJI_JOURNAL_INFO (0) LGANC 190
LGJI_LOG_TYPE (22) LGANC 190
LGJI_STATUS (24) LGANC 190
LGJI_STREAM (8) LGANC 190
LGJI_STREAM_TOKEN 190
LGJI_SYSTEM_LOG (23) LGANC 190
LGJMC_JNL_TEMPLATE_I (10) LGANC 191
LGJMC_JNL_TEMPLATE_X (8) LGANC 191
LGJMC_JOURNALMODEL_CONTENT (0) LGANC 191
LGJMC_JOURNALMODEL_NAME (0) LGANC 191
LGJMC_LOG_TYPE (32) LGANC 191
LGJMC_STREAM_PROTO (18) LGANC 191
LGSD_FAILED_LOG (1B) LGANC 189
LGSD_LOGBUF_TKN (24) LGANC 189

LGSD_STREAM (0) LGANC   189
LGSD_STREAM_DATA (0) LGANC   189
LGSD_STREAM_LOCK (20) LGANC   189
LGSD_STRUCTURE_NAME (28) LGANC   190
LGSD_SYSTEM_LOG (1A) LGANC   189
LGSD_USE_CT (1C) LGANC   189
LGSF   199
LGSL_BLOCK_HEADER (0) LGSF   199
LGSL_RECORD_HEADER (0) LGSF   200
LGUOW_CHAIN_HEAD (0) LGANC   191
LGUOW_CHAIN_NEXT (0) LGANC   191
LGUOW_FORCE_TOKEN (8) LGANC   191
LGUOW_HEADER (0) LGANC   191
LGUOW_STREAM_FORCE (0) LGANC   191
LGUOW_STREAM_TOKEN (4) LGANC   191
LGUOW_TIME_STAMP (4) LGANC   191
LI   306
LIBRARY_LOCK_NAME 8 LDCBS   175
life
  cics/db2 life of task block, D2LOT   93
LIFO   203
LIMIT_BLOCK_ID (8) L2BL   209
LINES_WRITTEN (816) STUCB   375
link
  recovery manager link class data, RMLK   305
  recovery manager link instance, RMLK   309
  recovery manager link set instance, RMLS   318
LINK (0) L2CH   221
LINK (0) L2SR   251
LINK_COMMIT 4 RMLK   309, 317
LINK_COMMIT_ABENDED   314, 319, 333
LINK_COMMITTED 4 RMLK   309, 317
LINK_FACTORY   306
LINK_FLAGS (4C) RMLK   310
LINK_FLAGS (954) RMLK   307
LINK_ID (5C) RMLK   317
LINK_ID (9B8) RMLK   308
LINK_ID (B0) RMLK   310
LINK_ID_SOURCE (17) RMLK   317
LINK_ID_SOURCE (6B) RMLK   310
LINK_ID_SOURCE (973) RMLK   308
LINK_ID_TYPE (0) RMLK   316
LINK_IN_DOUBT 4 RMLK   309, 317
LINK_R_COMMITTED 4 RMLK   309, 317
LINK_R_FORGET 4 RMLK   309, 317
LINK_R_PREPARE 4 RMLK   309, 317
LINK_R_REQUEST_COMMIT 4 RMLK   309, 317
LINK_RESET 4 RMLK   309, 317
LINK_ROLLBACK_NOT_SUPPORTED (BIT) RMLK   314
LINK_ROLLBACK_NOT_SUPPORTED (BIT) RMLS   319
LINK_ROLLBACK_NOT_SUPPORTED (BIT) RMUW   333
LINK_S_COMMITTED 4 RMLK   309, 317
LINK_S_PREPARE 4 RMLK   309, 317
LINK_S_REQUEST_COMMIT 4 RMLK   309, 317
LINK_SELECTED_LAST 4 RMLK   309, 317
LINK_STATISTICS   307
LINK_STATUS (50) RMLK   310
LINK_STATUS (958) RMLK   308
LINK_TOKEN (38) RMLK   309
LINK_TOKEN (940) RMLK   307
LINK_TOKENS (40) RMLK   305
LINKS   313, 332
LINKS_FORGOTTEN (BIT) RMLK   312
LINKS_FORGOTTEN (BIT) RMUW   331
LINKS_PRESENT (2D) RMLK   311
LINKS_PRESENT (2D) RMUW   330
LINKSET_CHAIN (28) RMLK   309
LINKSET_CHAIN (930) RMLK   307
list
  properties list, FEP12   135
LISTEN_PARMS (18) SOA   371
LL (0) TSAUX   388
LL (0) TSMN   392
LLBB (0) TSAUX   388
LLBB (0) TSMN   392
LLE (0) PGDCC   291
LLE_INSTANCE (C) PGDCC   291
LLE_NEXT (0) PGDCC   291
LLE_PPTE_ADDRESS (8) PGDCC   291
LLE_PREFIX (0) PGDCC   291
LLE_PREV (4) PGDCC   291
LM_ARROW (2) LMCB1   205
LM_BLOCK_NAME (8) LMCB1   205
LM_COMP_AND_SWAP_SECTION (18) LMCB1   205
LM_CS_COUNT (1A) LMCB1   205
LM_CS_MODE_S   205
LM_CS_NEXT_PTR (1C) LMCB1   205
LM_CS_OWNER (18) LMCB1   205
LM_DFH (3) LMCB1   205
LM_DOMID (6) LMCB1   205
LM_LENGTH (0) LMCB1   205
LM_LOCK_NAME (10) LMCB1   205
LM_LOCK_REQUESTS (24) LMCB1   205
LM_LOCK_SUSPENDS (28) LMCB1   205
LM_LOCK_TOKEN (20) LMCB1   205
LM_PREFIX (0) LMCB1   205
LMCB1   204
LMCB2   206
LMLM_LOCK_FREE 1 L2DM   225
LMLM_LOCK_FREE 1 RMDM   303
LMLM_LOCK_HELD 1 L2DM   225
LMLM_LOCK_HELD 1 RMDM   303
loader
  loader domain control blocks, LDCBS   164
LOADER_INITIALISING 2 LDCBS   175
LOADER_PRE_INITIALISED 2 LDCBS   175
LOADER_PRE_INITIALISING 2 LDCBS   175
LOADER_QUIESCED 2 LDCBS   175
LOADER_QUIESCING 2 LDCBS   175
LOADER_TERMINATED 2 LDCBS   175
LOADER_TERMINATING 2 LDCBS   175
LOADER_UP_AND_RUNNING 2 LDCBS   175
LOB (0) LDCBS   172
LOB_APE_CELL_POOL_SIZE   172
LOB_CREATION_STCK (10) LDCBS   172
LOB_CSECTL_CELL_POOL_SIZE (C) LDCBS   172
LOB_LLACOPY_STATUS (5) LDCBS   172
LOB_LPA_STATUS (4) LDCBS   172
LOB_STORAGE_FACTOR (0) LDCBS   172
local
  data tables local access anchor blocks, DTCPS   68
LOCAL_ACCESS_ID (16) RMUW   335
LOCAL_CATALOG 2 CCGD   31
LOCAL_COLD_LOG_RECORD 4 RMUW   336, 340
LOCAL_COMMIT_LOGGED (BIT) RMLK   312
LOCAL_COMMIT_LOGGED (BIT) RMUW   331
LOCAL_ME 1 CCGD   31
LOCAL_UOW_STATUS (9F2) RMLK   308
LOCAL_UOW_STATUS (EA) RMLK   311
LOCALLY_COMMITTED (BIT) RMLK   312
LOCALLY_COMMITTED (BIT) RMUW   331
locator
  file control locks locator block, FLLBC   150
lock
  lock manager domain anchor block, LMCB1   204
  lock manager domain quickcell headers, LMCB2   206
  log manager lock tracker class, L2LT   238
  temporary storage ownership lock class, TSOL   394
  temporary storage resource lock class, TSRL   401
  transaction manager resource lock element, XMRLC   440
LOCK_ADDED (C0) L2BS   213
LOCK_ADDED (C0) L2SR   244
LOCK_ELEMENT (0) LMCB1   205
LOCK_ERROR_CODE 4 DHANC   42
LOCK_ERROR_CODE 4 LGANC   193
LOCK_FAILED (BIT) DSANC   58
LOCK_MANAGEMENT (0) LMCB1   205
LOCK_STATUS (4) L2LT   238
LOCK_TOKEN (10) MEPS   257
LOCK_TOKEN (2C) L2BS   212
LOCK_TOKEN (2C) L2CH   220
LOCK_TOKEN (2C) L2SR   243
LOCK_TOKEN (48) STCB1   374
LOCK_WORDS (90) DSANC   55
LOCKING_INFO (10) MEPS   257
locks
  file control locks locator block, FLLBC   150
LOCKTOK (79C) DMCB1   47
LOCKTRACKER (0) L2LT   238
log
  log manager block class, L2BL   208
  log manager browseable stream class, L2BS   211
  log manager chain class, L2CH   219
  log manager hard stream class, L2HS   227
  log manager history point class, L2HP   226

log *(continued)*
  log manager l2dm class, L2DM 224
  log manager lock tracker class, L2LT 238
  log manager log formats, L2LF 231
  log manager record token class, L2RT 239
  log manager stream class, L2SR 242
  log manager system log class, L2SL 240
  log of logs failure record, LGFL 198
  recovery manager system log class data, RMSL 329
  recovery manager system log instance, RMSL 327
  system log format, LGSF 199
LOG (10) BAPT 23
LOG_DATA (14) CPCPS 33
LOG_DATA_BUFFER_LENGTH (10) CPCPS 33
LOG_DATA_BUFFER_PTR (30) CPCPS 32
LOG_DATA_EYECATCHER (2) CPCPS 33
LOG_DATA_HDR_LEN 2 CPCPS 33
LOG_DATA_LENGTH (2C) CPCPS 32
LOG_DATA_RECORD_LENGTH (0) CPCPS 33
LOG_NOT_DEFINED 251
LOG_NOT_DEFINED 4 L2HS 231
LOG_STREAM_STATS (1BC) L2BS 216
LOG_STREAM_STATS (1BC) L2SR 248
LOG_STREAM_STATS (CC) L2HS 230
LOG_TYPE (12A) L2BS 216
LOG_TYPE (12A) L2SR 248
LOG_TYPE (3A) L2HS 230
loggable
  recovery manager loggable object identity instance, RMLI 304
LOGGED_STATE (54) RMLK 310
LOGGED_STATE (95C) RMLK 308
logger
  logger domain anchor block, LGANC 188
  logger reusable extended iliffe vector class, RUEI 343
logic
  web business logic compatibility interface, WBA1C 413
  web business logic interface parameters, WBBLC 416
logname
  recovery manager logname class data, RMNM 320
  recovery manager logname instance, RMNM 321
  recovery manager logname set instance, RMNS 322
LOGNAME (1A) RMLK 317
LOGNAME (23) RMNM 321
LOGNAME (6E) RMLK 310
LOGNAME (976) RMLK 308
LOGON_PARMS (148) XCCBC 433
logs
  log of logs failure record, LGFL 198
LOGSTREAM_NAME 216, 248
LOGSTREAM_OPT_FIELDS 217, 249
LOGSTREAM_STATS 216, 248
LOGSTREAMTOKEN 250
LOST_ACCESS 251
LOST_ACCESS 4 L2BL 211
LOST_ACCESS 4 L2HS 231
LOST_DATA 251
LOST_DATA 4 L2BL 211
LOST_DATA 4 L2HS 231
LOST_DATA_WARNING (C5) L2BS 213
LOST_DATA_WARNING (C5) L2SR 244
LOT_ABEND_AD2S (BIT) D2LOT 94
LOT_ABEND_AD2T (BIT) D2LOT 94
LOT_ABEND_AD2U (BIT) D2LOT 94
LOT_ABEND_TXN_WITH_DUMP 1 D2LOT 95
LOT_ABORT_REQUEST 1 D2LOT 95
LOT_ACCOUNT_CLOCK (A8) D2LOT 94
LOT_ACCOUNT_LUNAME (A0) D2LOT 94
LOT_ACCOUNT_NETNAME (98) D2LOT 94
LOT_ACCOUNTING_TOKEN (98) D2LOT 94
LOT_ACEE_ADDRESS (38) D2LOT 93
LOT_ACQUIRE_LOCK_FAILED 1 D2LOT 95
LOT_API_CALL_IN_PROGRESS (BIT) D2LOT 94
LOT_API_DETACH (BIT) D2LOT 94
LOT_API_REQUEST_FAILED 1 D2LOT 95
LOT_APPL_MUST_ABORT (BIT) D2LOT 93
LOT_ATTACH_IN_STANDBY_MODE 1 D2LOT 95
LOT_ATTACH_SHUTDOWN_IN_PROGRESS 1 D2LOT 95
LOT_ATTACH_SUBTASK_FAILED 1 D2LOT 96
LOT_ATTACH_SUBTASK_NO_STORAGE 1 D2LOT 96
LOT_AUTH_TYPE_INVALID 1 D2LOT 95
LOT_CALL_PARMS (30) D2LOT 93
LOT_CALL_PARMS_HIGH 93
LOT_CICS_ABORT_DB2_COMMIT 1 D2LOT 95
LOT_CICS_SHUTDOWN_REQUEST 1 D2LOT 95
LOT_COMMIT_REQUEST 1 D2LOT 95
LOT_CONN_SUBTASK_ABEND 1 D2LOT 95
LOT_CREATE_THREAD_FAILED 1 D2LOT 95
LOT_CSUB (1C) D2LOT 93
LOT_CURRENT_REQUEST (5C) D2LOT 93
LOT_DB2_RESOLVE_INDOUBT_ABEND 1 D2LOT 95
LOT_DB2ENTRY_DISABLED 1 D2LOT 96
LOT_DEFERRED_ABENDS 94
LOT_DSNC_COMMAND_REQUEST 1 D2LOT 95
LOT_DSNC_COMMAND_REQUEST_FAILED 1 D2LOT 95
LOT_DYN_PLAN_ALLOWED (BIT) D2LOT 93
LOT_ECB (34) D2LOT 93
LOT_EDF_CALL_FAILED 1 D2LOT 95
LOT_END_OF_TASK_REQUEST 1 D2LOT 95
LOT_ERROR_CODES (69) D2LOT 94
LOT_ERROR_CODES_MINUS_ONE (6A) D2LOT 94
LOT_ERROR_CODES_MINUS_THREE (6C) D2LOT 94
LOT_ERROR_CODES_MINUS_TWO (6B) D2LOT 94
LOT_EYE (2) D2LOT 93
LOT_FRB (AE) D2LOT 94
LOT_GETMAIN_FAILED 1 D2LOT 96
LOT_GLB_TCB_READYQ (48) D2LOT 93
LOT_GWA_CHAIN_NEXT (20) D2LOT 93
LOT_GWA_CHAIN_PREV (24) D2LOT 93
LOT_IFI_API_BUT_MUST_ABORT 1 D2LOT 95
LOT_IFI_API_REQUEST 1 D2LOT 95
LOT_IFI_API_REQUEST_FAILED 1 D2LOT 95
LOT_IFI_EDF_REQUEST 1 D2LOT 95
LOT_INDOUBT_NEXT (50) D2LOT 93
LOT_INSTALLATION_ERROR 1 D2LOT 95
LOT_INVALID_DDLO_REASON 1 D2LOT 96
LOT_INVALID_DDLO_RESPONSE 1 D2LOT 96
LOT_INVALID_RMI_VERB 1 D2LOT 96
LOT_INVALID_THREAD_STATE 1 D2LOT 96
LOT_LEN (0) D2LOT 93
LOT_LEVEL1_TRACE (BIT) D2LOT 94
LOT_LEVEL2_TRACE (BIT) D2LOT 94
LOT_LOST_OUR_THREAD 1 D2LOT 96
LOT_MUST_ABORT 1 D2LOT 95
LOT_NO_THREAD 1 D2LOT 95
LOT_ONLY_DB2_INDOUBT 1 D2LOT 95
LOT_OVERFLOW_TO_POOL (BIT) D2LOT 93
LOT_PLAN_NAME (54) D2LOT 93
LOT_PREFIX (0) D2LOT 93
LOT_PREPARE_ABENDED 1 D2LOT 95
LOT_PREPARE_READ_ONLY (BIT) D2LOT 94
LOT_PREPARE_REQUEST 1 D2LOT 95
LOT_PRIMARY_AUTH_NAME (80) D2LOT 94
LOT_RCT_CHAIN_NEXT (28) D2LOT 93
LOT_RCT_CHAIN_PREV (2C) D2LOT 93
LOT_RCT_TAMPER_ERROR 1 D2LOT 95
LOT_RCTE (18) D2LOT 93
LOT_RCTE_READYQ (40) D2LOT 93
LOT_READ_ONLY_INDICATOR 94
LOT_READYQ_COUNT (44) D2LOT 93
LOT_READYQ_NEXT (40) D2LOT 93
LOT_RECOVERY_ROUTINE_ENTERED 1 D2LOT 95
LOT_RELEASE_LOCK_FAILED 1 D2LOT 95
LOT_REQUEST_FLAGS (60) D2LOT 93
LOT_REQUEST_INDICATORS (5C) D2LOT 93
LOT_REQUEST_MINUS_ONE (5D) D2LOT 93
LOT_REQUEST_MINUS_THREE (5F) D2LOT 93
LOT_REQUEST_MINUS_TWO (5E) D2LOT 93
LOT_RESYNC_FAILED_INITIAL_START 1 D2LOT 95
LOT_RESYNC_LOST_TO_INITIAL 1 D2LOT 95
LOT_RETURN_CODES (68) D2LOT 94
LOT_RMI_RETURN_CODE (68) D2LOT 94
LOT_RMI_RETURN_CODE_OK 1 D2LOT 95
LOT_ROLLBACK_TXN_FOR_DEADLOCK 1 D2LOT 95
LOT_SECONDARY_AUTH_NAME (88) D2LOT 94
LOT_SHUTDOWN_WHILE_COMMIT_ABORT 1 D2LOT 95
LOT_SIGNON_FAILED 1 D2LOT 95
LOT_SINGLE_PHASE_BACKED_OUT 1 D2LOT 95
LOT_SINGLE_PHASE_COMMIT 1 D2LOT 95
LOT_SINGLE_PHASE_COMMIT_FAILED 1 D2LOT 95
LOT_SPI_REQUEST 1 D2LOT 95
LOT_SQL_API_BUT_MUST_ABORT 1 D2LOT 95
LOT_SQL_API_REQUEST 1 D2LOT 95
LOT_SQL_API_REQUEST_FAILED 1 D2LOT 95
LOT_SQL_EDF_REQUEST 1 D2LOT 95
LOT_SQL_STATUS 94
LOT_SUBTASK_ABEND_REASON (90) D2LOT 94

LOT_SWAP_WORD 94
LOT_TASK_PURGED_FROM_CICS (BIT) D2LOT 94
LOT_TCA (14) D2LOT 93
LOT_TCB_READYQ_COUNT (4C) D2LOT 93
LOT_TCB_READYQ_NEXT (48) D2LOT 93
LOT_TERMINAL_TRANS (BIT) D2LOT 93
LOT_THREAD_RESOURCE_UNAVAILABLE 1 D2LOT 95
LOT_TRACE_FLAGS 94
LOT_TRANSID (10) D2LOT 93
LOT_TXNS_LAST_CALL (BIT) D2LOT 93
LOT_UNKNOWN_CALL 1 D2LOT 95
LOT_UNKNOWN_RMI_CALL 1 D2LOT 95
LOT_UR_SHOULD_NOT_BE_INDOUBT 1 D2LOT 95
LOT_UR_TOKEN 94
LOT_WAIT_MVS_FAILED 1 D2LOT 96
LOT_WLM_PERF_TOKEN (3C) D2LOT 93
LPA_NAME 5 LDCBS 175
LTE (0) SOA 368
LTE_ADDR (1B0) SOA 369
LTE_ARROW (2) SOA 368
LTE_ATTACH_COUNT (240) SOA 369
LTE_BLOCK_NAME (8) SOA 368
LTE_CID (248) SOA 369
LTE_CONNECTION_COUNT (40) SOA 368
LTE_CONNECTION_FAILURE (BIT) SOA 368
LTE_DEREGISTERING (BIT) SOA 368
LTE_DFH (3) SOA 368
LTE_DOMID (6) SOA 368
LTE_FLAG1 (54) SOA 368
LTE_FLAG2 (55) SOA 368
LTE_IDENTITY_NO (44) SOA 368
LTE_IMMCLOSING (BIT) SOA 368
LTE_INET_ADDR (1B0) SOA 369
LTE_LENGTH (0) SOA 368
LTE_LISTEN_BACKLOG (4C) SOA 368
LTE_NEW (BIT) SOA 368
LTE_NEXT (10) SOA 368
LTE_OPEN_TIME (230) SOA 369
LTE_OPEN_TIME_HIGH (230) SOA 369
LTE_OPEN_TIME_LOW (234) SOA 369
LTE_PEAK_CONN (244) SOA 369
LTE_PORT (18) SOA 368
LTE_PREFIX (0) SOA 368
LTE_PREV (14) SOA 368
LTE_READY_ECB (48) SOA 368
LTE_RECV_BYTES (228) SOA 369
LTE_RECV_BYTES_HIGH (228) SOA 369
LTE_RECV_BYTES_LOW (22C) SOA 369
LTE_RECV_COUNT (23C) SOA 369
LTE_RECV_TIMEOUT 368
LTE_SEND_BYTES (220) SOA 369
LTE_SEND_BYTES_HIGH (220) SOA 369
LTE_SEND_BYTES_LOW (224) SOA 369
LTE_SEND_COUNT (238) SOA 369
LTE_SERVER_ADDRESS_AREA 368
LTE_SERVER_BIN_IP_ADDR (168) SOA 368
LTE_SERVER_HOSTNAME_BUF (58) SOA 368
LTE_SERVER_HOSTNAME_LEN (158) SOA 368
LTE_SERVER_IP_ADDRESS (159) SOA 368
LTE_SERVICE_AREA (16C) SOA 368
LTE_SERVICE_CLIAUTH (BIT) SOA 368
LTE_SERVICE_FLAGS (18C) SOA 368
LTE_SERVICE_NAME (16C) SOA 368
LTE_SERVICE_SSL (BIT) SOA 368
LTE_SERVICE_TRANID (17C) SOA 368
LTE_SERVICE_TSQPREFIX (180) SOA 368
LTE_SERVICE_URM (174) SOA 368
LTE_SOCKADDR (1AE) SOA 369
LTE_SOCKADDR_HEADER (1AE) SOA 369
LTE_SOCKET (50) SOA 368
LTE_SOCKET_BOUND (BIT) SOA 368
LTE_SOCKET_CLOSED (BIT) SOA 368
LTE_SOCKET_CREATED (BIT) SOA 368
LTE_SOCKET_GETCLID (BIT) SOA 368
LTE_SOCKET_LISTENED (BIT) SOA 368
LTE_STATISTICS_DATA (220) SOA 369
LTE_STE_CHAIN 368
LTE_STE_EMPTY_ECB (1C) SOA 368
LTE_STE_HEAD (24) SOA 368
LTE_STE_NUM_ENTRIES (20) SOA 368
LTE_UNIX_ADDR 369
LTE_WLM_CRITIAL (BIT) SOA 369
LTE_WLM_DATA (190) SOA 368
LTE_WLM_FLAGS (191) SOA 369
LTE_WLM_GROUPNAME (19C) SOA 369
LTE_WLM_RETCODE (194) SOA 369
LTE_WLM_RSNCODE (198) SOA 369
LTE_WLM_STATE (190) SOA 368

M

macro
  macro save area, PGA 285
macro-compatability
  sm macro-compatability anchor block, SMMCC 364
MAFPB (0) MNAFB 260
MAFPB_ARROW (2) MNAFB 260
MAFPB_BLOCK_ID (8) MNAFB 260
MAFPB_CREATION_STCK (3C) MNAFB 261
MAFPB_DFH (3) MNAFB 260
MAFPB_DOMAIN (6) MNAFB 260
MAFPB_FUNCTION (10) MNAFB 260
MAFPB_GTF_TRACE_FLAG 260
MAFPB_GTF_TRACE_OFF 0 MNAFB 261
MAFPB_GTF_TRACE_ON 0 MNAFB 261
MAFPB_ID_STRING 8 MNAFB 261
MAFPB_INVALID_FUNCTION 1 MNAFB 261
MAFPB_INVALID_PB_TOKEN 1 MNAFB 261
MAFPB_INVALID_RECORD_LENGTH 1 MNAFB 261
MAFPB_LENGTH (0) MNAFB 260
MAFPB_NO_AUTHORISATION 1 MNAFB 261
MAFPB_NO_FESTAE 1 MNAFB 261
MAFPB_NO_STORAGE_253 1 MNAFB 261
MAFPB_NO_STORAGE_HASH 1 MNAFB 261
MAFPB_NO_STORAGE_HASH_ELEM 1 MNAFB 261
MAFPB_NO_STORAGE_MNACB 1 MNAFB 261
MAFPB_NO_STORAGE_SMF 1 MNAFB 261
MAFPB_NOT_CICS_RECORD 1 MNAFB 261
MAFPB_OK 1 MNAFB 261
MAFPB_PREFIX (0) MNAFB 260
MAFPB_RESPONSE (12) MNAFB 260
MAFPB_RTNREG0 260
MAFPB_RTNREG1 (24) MNAFB 260
MAFPB_RTNREG15 (28) MNAFB 260
MAFPB_SMF_ERROR 1 MNAFB 261
MAFPB_SMF_RC (1C) MNAFB 260
MAFPB_SMF_RECORD 260
MAFPB_SMFEWTM 2 MNAFB 261
MAFPB_SYSEVENT 2 MNAFB 261
MAFPB_SYSEVENT_ERROR 1 MNAFB 261
MAFPB_SYSEVENT_RC (1D) MNAFB 260
MAFPB_SYSEVENT_RECORD (18) MNAFB 260
MAFPB_WLM_CONNECT 2 MNAFB 261
MAFPB_WLM_CONNECT_FAILED 1 MNAFB 261
MAFPB_WLM_CONNECT_TOKEN (2C) MNAFB 261
MAFPB_WLM_DISCONNECT 2 MNAFB 261
MAFPB_WLM_DISCONNECT_FAILED 1 MNAFB 261
MAFPB_WLM_NOTIFY 2 MNAFB 261
MAFPB_WLM_NOTIFY_FAILED 1 MNAFB 261
MAFPB_WLM_OP_OUT_OF_SEQUENCE 1 MNAFB 261
MAFPB_WLM_PB_CREATE 2 MNAFB 261
MAFPB_WLM_PB_CREATE_FAILED 1 MNAFB 261
MAFPB_WLM_PB_DELETE 2 MNAFB 261
MAFPB_WLM_PB_DELETE_FAILED 1 MNAFB 261
MAFPB_WLM_PERFORMANCE_BLOCK (30) MNAFB 261
MAFPB_WLM_REPORT 2 MNAFB 261
MAFPB_WLM_REPORT_FAILED 1 MNAFB 261
MAFPB_WLM_TRAN_END_TIME (34) MNAFB 261
main
  temporary storage main class, TSMN 392
management
  recovery manager domain management instance, RMDM 301
manager
  adapter resource manager, FEP02 113
  directory manager building blocks, DDBSC 35
  directory manager structures, DDCBC 36
  domain manager anchor block, DMCB1 47
  domain manager browse cursor, DMCB2 49
  domain manager enf state, DMENC 52
  domain manager wait queue element, DMCB3 50
  handle manager declarations, PGHM 293
  lock manager domain anchor block, LMCB1 204
  lock manager domain quickcell headers, LMCB2 206
  log manager block class, L2BL 208

manager *(continued)*
  log manager browseable stream class, L2BS 211
  log manager chain class, L2CH 219
  log manager hard stream class, L2HS 227
  log manager history point class, L2HP 226
  log manager l2dm class, L2DM 224
  log manager lock tracker class, L2LT 238
  log manager log formats, L2LF 231
  log manager record token class, L2RT 239
  log manager stream class, L2SR 242
  log manager system log class, L2SL 240
  parameter manager domain anchor block, PAA 283
  program manager control blocks, PGDCC 286
  recovery manager domain management instance, RMDM 301
  recovery manager identity instance, RMID 303
  recovery manager link class data, RMLK 305
  recovery manager link instance, RMLK 309
  recovery manager link set instance, RMLS 318
  recovery manager loggable object identity instance, RMLI 304
  recovery manager logname class data, RMNM 320
  recovery manager logname instance, RMNM 321
  recovery manager logname set instance, RMNS 322
  recovery manager resource owner instance, RMRO 324
  recovery manager system log class data, RMSL 329
  recovery manager system log instance, RMSL 327
  recovery manager unit of work class data, RMUW 337
  recovery manager unit of work instance, RMUW 330
  storage manager anchor block, SMDCC 345
  transaction manager catalog records, XMCAT 438
  transaction manager domain anchor block, XMANC 435
  transaction manager resource lock element, XMRLC 440
  transaction manager tran. browse element, XMXBC 441
  transaction manager transaction class, XMCLC 439
  transaction manager transaction definition, XMXDC 441
  transaction manager transaction, XMXNC 445
  web state manager data, WBSTC 422
map
  tsf - eye catcher map, FEP09 131
MARK (9EF) RMLK 308
MARK (E7) RMLK 311
MASTER_PREV (0) L2LF 233
MASTER_PREV (10) L2LF 235
MASTER_PREV (10) LGSF 200
MASTERCHAINHEADER (0) L2LF 233
MAX_BLOCK_SIZE 216, 230, 248
MAX_CICS24_SAA_LENGTH 4 SMMCC 366
MAX_DATA_LENGTH (38) CCGD 29
MAX_DSA_LIMIT 4 SMDCC 362
MAX_EDSA_LIMIT 4 SMDCC 362
MAX_INSERTS 1 MEPS 259
MAX_LANGUAGES 1 MEPS 259
MAX_LIOA_LENGTH 4 SMMCC 366
MAX_QUEUES 4 MEMMS 256
MAX_REC_LEN 208
MAX_REPLIES 1 MEPS 259
MAX_ROUTE_CODES 4 MEMMS 256
MAX_SECONDARY_ABOVE 4 SMDCC 363
MAX_SECONDARY_BELOW 4 SMDCC 363
MAX_SHARED_CICS24_SAA_LENGTH 4 SMMCC 366
MAX_SYMPTOM_STRING_LEN 4 MEPS 259
MAX_TIOA_LENGTH 4 SMMCC 366
MAX_TRACE_BLOCK_LEN 4 L2HS 231
MAX_TUNING_INTERVALS 4 SMDCC 363
MAXDSA 4 SMDCC 363
MAXDSAS 4 LDCBS 175
MAXIMUM_WAIT_INTERVAL (20) DSANC 54
MAXIMUM_WAIT_INTERVAL_SIT (84) DSANC 54
MAXITEMLENGTH 4 TSQU 398
MAXITEMS 4 TSQU 398
MAXKEYLEN 4 DDCBC 37
MAXOPENTCBS (658) DSANC 58
MAXPOOL 4 SMDCC 363
MAXSUBPOOLS 4 LDCBS 174
MB16 4 SMDCC 356
MBR 390
MBR_MBRHEAD (0) TSMN 390
MBR_NEXT (0) TSMN 390
MBR_PREFIX (18) TSMN 391
MBR_PREV (4) TSMN 391
MBR_TRANID (8) TSMN 391
MBR_TRANNUM (C) TSMN 391
MBR_TRANTOKEN (10) TSMN 391
MCA (0) SMMCC 364
MCA_ARROW (2) SMMCC 364
MCA_BLOCK_NAME (8) SMMCC 364
MCA_CONTROL_SPID 364
MCA_CONTROL_SPTOKEN (54) SMMCC 364
MCA_CONTROL_SPTOKEN_P (54) SMMCC 364
MCA_DFH (3) SMMCC 364
MCA_DOMID (6) SMMCC 364
MCA_LENGTH (0) SMMCC 364
MCA_PREFIX (0) SMMCC 364
MCA_SHARED_SPID 364
MCA_SHARED_SPTOKEN (48) SMMCC 364
MCA_SHARED_SPTOKEN_P (48) SMMCC 364
MCA_SHRC24_SPID 364
MCA_SHRC24_SPTOKEN 364
MCA_SHRC24_SPTOKEN_P (18) SMMCC 364
MCA_SHRC31_SPID 364
MCA_SHRC31_SPTOKEN (30) SMMCC 364
MCA_SHRC31_SPTOKEN_P (30) SMMCC 364
MCA_SHRU24_SPID 364
MCA_SHRU24_SPTOKEN (24) SMMCC 364
MCA_SHRU24_SPTOKEN_P (24) SMMCC 364
MCA_SHRU31_SPID 364
MCA_SHRU31_SPTOKEN (3C) SMMCC 364
MCA_SHRU31_SPTOKEN_P (3C) SMMCC 364
MCA_SMMC_ACTIVE 365
MCA_SUBPOOLS 364
MCA_TP_SPID 365
MCA_TP_SPTOKEN (6C) SMMCC 365
MCA_TP_SPTOKEN_P (6C) SMMCC 365
MCA_TP24_SPID 364
MCA_TP24_SPTOKEN (60) SMMCC 364
MCA_TP24_SPTOKEN_P (60) SMMCC 364
MDA 390
MDA_DEFAULT_MDBP (28) TSMN 390
MDA_EYECATCHER (0) TSMN 390
MDA_EYECATCHER_STRING 391
MDA_MBR_FIRST (20) TSMN 390
MDA_MBR_LAST (24) TSMN 390
MDA_MBR_SPTOKEN (10) TSMN 390
MDA_MBRHEAD (20) TSMN 390
MDA_MDB_FIRST (18) TSMN 390
MDA_MDB_LAST (1C) TSMN 390
MDA_MDB_SPTOKEN (8) TSMN 390
MDA_MDBHEAD (18) TSMN 390
MDB 390
MDB_DEFAULT (BIT) TSMN 390
MDB_FLAGS (54) TSMN 390
MDB_MAIN (BIT) TSMN 390
MDB_MASKED_PREFIX (40) TSMN 390
MDB_MDBHEAD (0) TSMN 390
MDB_NAME (8) TSMN 390
MDB_NEXT (0) TSMN 390
MDB_POOL_NAME (58) TSMN 390
MDB_POOL_TOKEN (60) TSMN 390
MDB_PREFIX (20) TSMN 390
MDB_PREFIX_MASK (30) TSMN 390
MDB_PREFIXLEN (50) TSMN 390
MDB_PREV (4) TSMN 390
MDB_QNAME (10) TSMN 390
MDB_RECOVERABLE (BIT) TSMN 390
MDB_REMOTE_PREFIX (68) TSMN 390
MDB_SECURITY (BIT) TSMN 390
MDB_SYSID (64) TSMN 390
MDL_CATALOG_ERROR 391
MDL_DISASTER 391
MDL_DUPLICATE_NAME 391
MDL_DUPLICATE_PREFIX 391
MDL_END_BROWSE 391
MDL_INVALID_BROWSE_TOKEN 391
MDL_INVALID_NAME 391
MDL_INVALID_PREFIX 391
MDL_NOT_FOUND 391
MDL_OK 391
MDL_PURGED 391
MDL_RESPONSE (0) TSMN 391
ME_DOMAIN_STATUS (1D) MEPS 257
ME_GLOBAL_CAT 1 LDCBS 177
ME_LOCAL_CAT 1 LDCBS 177
MECR_DEFAULT_LANGUAGE (26) MEPS 258
MECR_DEFAULT_LANGUAGE_CODE (27) MEPS 258
MECR_LANGUAGES_USED (2) MEPS 258
MECR_MESSAGE_CASE (0) MEPS 258
MECR_MSG_LEVEL (BIT) MEPS 258

MECR_NUMBER_OF_LANGS (1) MEPS 258
MEID_BADSTCK 4 TIA 379
MEID_LESSTHAN_PARAMETER 4 PAA 284
MEID_LOOP 4 PAA 284
MEID_LOOP 4 TIA 379
MEID_RECOV 4 TIA 379
MEID_RECOVERY 4 PAA 284
MEID_SEVERE_ERROR 4 PAA 284
MEMMS 252
MEPS 257
message
  message domain anchor block, MEPS 257
  message table definition, MEMMS 252
MESSAGE_CASE 257
MESSAGE_DEST 1 MEMMS 256
MESSAGE_IDENT 1 MEMMS 256
MESSAGE_INFO 257
MESSAGE_RCS 1 MEMMS 256
MESSAGE_TDQS 1 MEMMS 256
MESSAGE_TEMPLATE 1 MEMMS 256
MET_HEADER_LENGTH (0) MEMMS 252
MET_MODULE_HEADER (0) MEMMS 252
METG_AREA_LENGTH (0) MEMMS 252
METG_DATE_FORMAT (2) MEMMS 252
METG_DECIMAL_FORMAT (18) MEMMS 252
METG_MESSAGE_GLOBALS (0) MEMMS 252
METG_NEGNO_FORMAT (15) MEMMS 252
METG_NUMERIC_SET (1F) MEMMS 252
METG_REPLY_FOLD (29) MEMMS 252
METG_TIME_FORMAT (C) MEMMS 252
METH_ARROW (1) MEMMS 252
METH_ASMDATE (16) MEMMS 252
METH_ASMTIME (1F) MEMMS 252
METH_AT_SYMBOL (1E) MEMMS 252
METH_MODULE_IDENT (2) MEMMS 252
METH_PTFLEVEL (E) MEMMS 252
METH_RELEASE (A) MEMMS 252
METM_APPLID (BIT) MEMMS 253
METM_ARROW (1) MEMMS 252
METM_ASMDATE (16) MEMMS 252
METM_ASMTIME (1F) MEMMS 253
METM_AT_SYMBOL (1E) MEMMS 253
METM_COMPONENT_ID (2) MEMMS 253
METM_CONSOLE (BIT) MEMMS 253
METM_DATE (BIT) MEMMS 253
METM_DEST_TYPES (2) MEMMS 253
METM_ELEM_DATA (1) MEMMS 254
METM_ELEMENT (0) MEMMS 254
METM_ELEMENT_TYPE (0) MEMMS 254
METM_EXIT_DATA (2) MEMMS 255
METM_EXIT_ELEMS 255
METM_EXIT_FORMAT (3) MEMMS 255
METM_EXIT_MAP (0) MEMMS 255
METM_EXIT_TYPE (2) MEMMS 255
METM_HEADER (0) MEMMS 252
METM_HEADER_LENGTH (0) MEMMS 252
METM_INSERT_ELEMENT (0) MEMMS 254
METM_INSERT_FORMAT (2) MEMMS 254
METM_INSERT_ID 254
METM_MESSAGE_CODES (6) MEMMS 253
METM_MESSAGE_COMPONENT (0) MEMMS 253
METM_MESSAGE_DEFN (0) MEMMS 253
METM_MESSAGE_IDENT (0) MEMMS 253
METM_MESSAGE_NO (4) MEMMS 253
METM_MODULE_IDENT (2) MEMMS 252
METM_MSG_COMPONENT_TYPE (0) MEMMS 253
METM_MSG_DESTINATIONS (0) MEMMS 253
METM_MSG_RCS (0) MEMMS 254
METM_MSG_TDQS (0) MEMMS 254
METM_MSG_TEMPLATE (0) MEMMS 254
METM_MSGDEF_LENGTH 253
METM_MSGDESTS_LENGTH 253
METM_MSGENTRY_LENGTH (3) MEMMS 253
METM_MSGIDENT_LENGTH 253
METM_NETNAME (BIT) MEMMS 253
METM_NOREROUTE (A) MEMMS 253
METM_OPERATOR_ACTION (6) MEMMS 253
METM_OPTINS_IDENT (0) MEMMS 254
METM_OPTINS_LENGTH (1) MEMMS 254
METM_OPTINS_TEXT (2) MEMMS 254
METM_OPTIONAL_INSERT (0) MEMMS 254
METM_OPTVALUES_COUNT (3) MEMMS 254
METM_OPTVALUES_DATA (3) MEMMS 254
METM_PRIMAB (BIT) MEMMS 253
METM_PROGNAME (BIT) MEMMS 253
METM_PTFLEVEL (E) MEMMS 252
METM_RC_DATA (2) MEMMS 254
METM_RC_ELEMS 254
METM_RELEASE (A) MEMMS 252
METM_REPLY_ELEMENT (0) MEMMS 254
METM_REPLY_IDENT 254
METM_REPLY_LENGTH (2) MEMMS 254
METM_REPLY_TEXT (3) MEMMS 254
METM_RESP2_VALUE (8) MEMMS 253
METM_SECAB (BIT) MEMMS 253
METM_SEVERITY (7) MEMMS 253
METM_SPECIAL_INSERT_ELEMENT (0) MEMMS 254
METM_SPECIAL_INSERT_ELEMS 254
METM_SPECIAL_INSERT_FORMAT (2) MEMMS 255
METM_SPECINS_GEN (8) MEMMS 253
METM_SPECINS_INDICATOR 253
METM_SPECINS_PC 253
METM_SPECINS_TM 253
METM_SYMPTOM (0) MEMMS 255
METM_SYMPTOM_DATA (2) MEMMS 255
METM_SYMPTOM_DATA_TYPE (1) MEMMS 255
METM_SYMPTOM_ELEMS 255
METM_SYMPTOM_INSERT_DATA (0) MEMMS 255
METM_SYMPTOM_INSERT_OFFSET (2) MEMMS 255
METM_SYMPTOM_SPECIAL_DATA (0) MEMMS 255
METM_SYMPTOM_SPECIAL_TYPE (2) MEMMS 255
METM_SYMPTOM_TEXT_DATA (0) MEMMS 255
METM_SYMPTOM_TEXT_LENGTH (2) MEMMS 255
METM_SYMPTOM_TEXT_STRING (3) MEMMS 255
METM_SYMPTOM_TYPE (0) MEMMS 255
METM_SYMSTRING 253
METM_SYMSTRING_DEFINITION (0) MEMMS 255
METM_SYMSTRING_DEFINITION_DATA (2) MEMMS 255
METM_SYSID (BIT) MEMMS 253
METM_SYSPRINT (BIT) MEMMS 253
METM_TDQ (BIT) MEMMS 253
METM_TDQ_DATA (2) MEMMS 254
METM_TDQ_ELEMS 254
METM_TEMPLATE_DATA (2) MEMMS 254
METM_TEMPLATE_ELEMS 254
METM_TERMCDBC (BIT) MEMMS 253
METM_TERMENDU (BIT) MEMMS 253
METM_TERMID (BIT) MEMMS 253
METM_TEXT_EL_LENGTH 254
METM_TEXT_ELEMENT (0) MEMMS 254
METM_TEXT_STRING (2) MEMMS 254
METM_TIME (BIT) MEMMS 253
METM_TRANID (BIT) MEMMS 253
METM_TRANNUM (BIT) MEMMS 253
METM_USER_EXIT_OFFSET (5) MEMMS 253
METM_USERID (BIT) MEMMS 253
METX_ENTRY1_OFFSET (6) MEMMS 252
METX_INDEX_DATA 252
METX_INDEX_ENTRIES (5) MEMMS 252
METX_INDEX_ENTRY (0) MEMMS 252
METX_INDEX_LENGTH (0) MEMMS 252
METX_MESSAGE_INDEX (0) MEMMS 252
METX_MESSAGE_PREFIX (2) MEMMS 252
METX_MSGSET_ADDRESS 252
METX_MSGSET_NAME (0) MEMMS 252
MIDDLE_END (80) DSTSK 66
MIN_DSA_LIMIT 4 SMDCC 362
MIN_EDSA_LIMIT 4 SMDCC 362
MIN_FIXED_LENGTH 4 SMDCC 362
MIN_PRIMARY_SIZE 4 SMDCC 363
MIN_SECONDARY_SIZE 4 SMDCC 363
MINKEYLEN 4 DDCBC 37
MIXED 1 MEPS 259
MIXED_CASE (BIT) PAA 283
MN_DUMP_ABEND_PROGRAM_CHECK 8 MNCBS 274
MN_DUMP_INSUFFICIENT_STORAGE 8 MNCBS 274
MN_DUMP_POSSIBLE_LOOP 8 MNCBS 274
MN_DUMP_SEVERE_ERROR 8 MNCBS 274
MN_DUMP_STORE_CLOCK_ERROR 8 MNCBS 274
MNA (0) MNCBS 269
MNA_ARROW (2) MNCBS 269
MNA_BLOCK_ID (8) MNCBS 269
MNA_CC_ERROR_FOUND 270
MNA_CC_UPDATE_REQUIRED (BIT) MNCBS 270
MNA_CONNECTOR_LENGTH (A4) MNCBS 271
MNA_CONNECTORS_LENGTH (A8) MNCBS 271

MNA_CONTROL_POOL (18) MNCBS 270
MNA_CONVERSE_NO 0 MNCBS 274
MNA_CONVERSE_STATUS (BIT) MNCBS 270
MNA_CONVERSE_YES 0 MNCBS 274
MNA_CPU_START_REQUIRED 1 MNCBS 274
MNA_CPU_STARTED 1 MNCBS 274
MNA_CPU_STOP_REQUIRED 1 MNCBS 274
MNA_CPU_STOPPED 1 MNCBS 274
MNA_CPU_TIMING 270
MNA_CR (E8) MNCBS 272
MNA_CURRENT_TMAS (2C) MNCBS 270
MNA_DATA_CLASS (7C) MNCBS 271
MNA_DATA_LENGTH (78) MNCBS 271
MNA_DFH (3) MNCBS 269
MNA_DFHMCT 8 MNCBS 273
MNA_DICTIONARY_CLASS 2 MNCBS 273
MNA_DICTIONARY_ENTRIES (8C) MNCBS 271
MNA_DICTIONARY_LENGTH (90) MNCBS 271
MNA_DICTIONARY_PTR (94) MNCBS 271
MNA_DICTIONARY_REQUIRED (BIT) MNCBS 270
MNA_DICTIONARY_USER_ENTRIES (98) MNCBS 271
MNA_DOMAIN (6) MNCBS 269
MNA_DOMAIN_STATUS (10) MNCBS 270
MNA_EXCEPTION_CLASS 2 MNCBS 273
MNA_EXCEPTION_OFF 0 MNCBS 274
MNA_EXCEPTION_ON 0 MNCBS 274
MNA_EXCEPTION_RECORD (54) MNCBS 271
MNA_EXCEPTION_RECORDS (12C) MNCBS 272
MNA_EXCEPTION_RECORDS_SUPP (130) MNCBS 272
MNA_EXCEPTION_STATUS (BIT) MNCBS 270
MNA_EXIT_POINT 8 MNCBS 273
MNA_FIP_NO 4 MNCBS 274
MNA_FIP_YES 4 MNCBS 274
MNA_FREQUENCY (D0) MNCBS 271
MNA_FREQUENCY_IN_PROGRESS (DC) MNCBS 272
MNA_FREQUENCY_OFF 4 MNCBS 274
MNA_FREQUENCY_TOKEN (D4) MNCBS 272
MNA_ID_STRING 8 MNCBS 273
MNA_LAST_RESET_TIME (150) MNCBS 272
MNA_LAST_SMF_RC (82) MNCBS 271
MNA_LAST_SYSEVENT_RC (83) MNCBS 271
MNA_LENGTH (0) MNCBS 269
MNA_LOAD_MCT_NAME (4C) MNCBS 270
MNA_LOAD_MCT_SUFFIX 270
MNA_MAFPB_PTR (128) MNCBS 272
MNA_MCT_ADDRESS (40) MNCBS 270
MNA_MCT_DELETE (BIT) MNCBS 270
MNA_MCT_FIELDS_EXCLUDED 270
MNA_MCT_INITIALISED (BIT) MNCBS 270
MNA_MCT_LENGTH (48) MNCBS 270
MNA_MCT_LOAD_ADDRESS (44) MNCBS 270
MNA_MCT_LOADED (BIT) MNCBS 270
MNA_MCT_NAME (38) MNCBS 270
MNA_MCT_SUFFIX 270
MNA_MONITORING_OFF 0 MNCBS 274
MNA_MONITORING_ON 0 MNCBS 274
MNA_MONITORING_STATUS (BIT) MNCBS 270
MNA_NO 0 MNCBS 274
MNA_OFF 0 MNCBS 274
MNA_ON 0 MNCBS 274
MNA_OUT_CONNECTORS (A0) MNCBS 271
MNA_OUT_CONNECTORS_PTR (9C) MNCBS 271
MNA_PA_ERROR_FOUND (BIT) MNCBS 270
MNA_PB_LENGTH_LEFT (60) MNCBS 271
MNA_PB_NEXT_FREE (64) MNCBS 271
MNA_PB_SIZE (58) MNCBS 271
MNA_PD_LENGTH (6C) MNCBS 271
MNA_PD_RECORDS (68) MNCBS 271
MNA_PERFORMANCE_BUFFER (5C) MNCBS 271
MNA_PERFORMANCE_CLASS 2 MNCBS 273
MNA_PERFORMANCE_OFF 0 MNCBS 274
MNA_PERFORMANCE_ON 0 MNCBS 274
MNA_PERFORMANCE_RECORD (70) MNCBS 271
MNA_PERFORMANCE_RECORDS (134) MNCBS 272
MNA_PERFORMANCE_RECORDS_SUPP (138) MNCBS 272
MNA_PERFORMANCE_STATUS (BIT) MNCBS 270
MNA_RECORD_ADDRESS (74) MNCBS 271
MNA_RECORD_TYPE_CONVERSE 4 MNCBS 273
MNA_RECORD_TYPE_DELIVER 4 MNCBS 273
MNA_RECORD_TYPE_FREQUENCY 4 MNCBS 273
MNA_RECORD_TYPE_SYNCPOINT 4 MNCBS 273
MNA_RECORD_TYPE_TERMINATE 4 MNCBS 273
MNA_SMF_BUFFER (84) MNCBS 271
MNA_SMF_ERRORS (140) MNCBS 272
MNA_SMF_RECORDS (13C) MNCBS 272
MNA_STATE_LOCK (28) MNCBS 270
MNA_STATUS_FLAGS 270
MNA_SUBSYSTEM_ID (E0) MNCBS 272
MNA_SUBSYSTEM_NAME 4 MNCBS 273
MNA_SYNCPOINT_NO 0 MNCBS 274
MNA_SYNCPOINT_STATUS (BIT) MNCBS 270
MNA_SYNCPOINT_YES 0 MNCBS 274
MNA_SYSEVENT_ERRORS (148) MNCBS 272
MNA_SYSEVENT_OFF 0 MNCBS 274
MNA_SYSEVENT_ON 0 MNCBS 274
MNA_SYSEVENT_RECORD (88) MNCBS 271
MNA_SYSEVENT_RECORDS (144) MNCBS 272
MNA_SYSEVENT_RETRIES (14C) MNCBS 272
MNA_SYSEVENT_STATUS (BIT) MNCBS 270
MNA_TIME (BIT) MNCBS 270
MNA_TIME_GMT 0 MNCBS 274
MNA_TIME_LOCAL 0 MNCBS 274
MNA_TMA_CELL_POOL (20) MNCBS 270
MNA_TMA_LENGTH (30) MNCBS 270
MNA_TMA_USER_AREA_LENGTH (34) MNCBS 270
MNA_USER_EXIT_STATUS 270
MNA_WLM_CONNECT_TOKEN (AC) MNCBS 271
MNA_WLM_CUR_SYS_PERFORMANCE_BLKS (C8) MNCBS 271
MNA_WLM_CURRENT_PERFORMANCE_BLKS (C0) MNCBS 271
MNA_WLM_DISABLED 0 MNCBS 274
MNA_WLM_ENABLED 0 MNCBS 274
MNA_WLM_FREE_PERFORMANCE_BLK (B8) MNCBS 271
MNA_WLM_MAX_PERFORMANCE_BLKS (BC) MNCBS 271
MNA_WLM_MAX_SYS_PERFORMANCE_BLKS (C4) MNCBS 271
MNA_WLM_NOTIFIED_MXT_VALUE (CC) MNCBS 271
MNA_WLM_PB_ARRAY_PTR (B0) MNCBS 271
MNA_WLM_PB_ARRAY_SIZE (B4) MNCBS 271
MNA_WLM_STATUS (BIT) MNCBS 270
MNA_YES 0 MNCBS 274
MNAFB 260
MNCBS 262
MNCR_CONVERSE_STATUS (BIT) MNCBS 273
MNCR_EXCEPTION_STATUS (BIT) MNCBS 272
MNCR_FLAGS (2) MNCBS 272
MNCR_FREQUENCY 273
MNCR_MCT_SUFFIX (0) MNCBS 272
MNCR_MONITORING_STATUS (BIT) MNCBS 273
MNCR_PERFORMANCE_STATUS (BIT) MNCBS 272
MNCR_SUBSYSTEM_ID (7) MNCBS 273
MNCR_SYNCPOINT_STATUS (BIT) MNCBS 273
MNCR_SYSEVENT_STATUS (BIT) MNCBS 273
MNCR_TIME (BIT) MNCBS 273
MNME_ABEND_PROGRAM_CHECK 4 MNCBS 274
MNME_CATALOGUE_READ_ERROR 4 MNCBS 274
MNME_CATALOGUE_UPDATE_ERROR 4 MNCBS 274
MNME_INSUFFICIENT_STORAGE 4 MNCBS 274
MNME_MCT_NOT_FOUND 4 MNCBS 274
MNME_MCT_NOT_FOUND_IN_LIBRARY 4 MNCBS 274
MNME_MONITORING_ACTIVE 4 MNCBS 274
MNME_MONITORING_INACTIVE 4 MNCBS 274
MNME_POSSIBLE_LOOP 4 MNCBS 274
MNME_SEVERE_ERROR 4 MNCBS 274
MNME_SMF_ERROR 4 MNCBS 274
MNME_STORE_CLOCK_ERROR 4 MNCBS 274
MNME_SYSEVENT_ERROR 4 MNCBS 274
MNME_SYSEVENT_RETRY 4 MNCBS 274
MNME_USING_DEFAULT_MCT 4 MNCBS 274
MNME_USING_MCT 4 MNCBS 274
MNO_ABEND 1 USANC 406
MNO_ABEND 1 XSANC 449
MNO_ABEND 4 LGANC 192
MNO_ABEND 4 SMDCC 362
MNO_ABEND 4 TSA 381
MNO_APPCLU_RACLIST_FAILED 4 XSANC 449
MNO_DOM_INIT_END 4 LGANC 192
MNO_DOM_INIT_START 4 LGANC 192
MNO_DSA_LIMIT 4 SMDCC 362
MNO_DSA_SIZE 4 SMDCC 362
MNO_EDSA_LIMIT 4 SMDCC 362
MNO_ENQ_LIMIT_EXCEEDED 1 USANC 406
MNO_EXIT_REJECTED_DEFINE 4 LGANC 192
MNO_FAQE_ERROR 4 SMDCC 362
MNO_FORCE_PURGE_REJECTED 4 RMUW 337, 340
MNO_FORMATTING_DATASET 4 TSA 381
MNO_INCOMPLETE_UOW_ERROR 4 RMUW 336, 341
MNO_INITIALISATION_ENDED 4 TSA 381

MNO_INITIALISATION_STARTED 4 TSA 381
MNO_INVALID_RDO_SWITCH 4 TSA 381
MNO_JNL_CATLG_DEL_FAIL 4 LGANC 192
MNO_JNL_CATLG_FAIL 4 LGANC 192
MNO_JNL_CONN_FAIL 4 LGANC 192
MNO_JNL_DEFINED 4 LGANC 192
MNO_JNL_DISCARDED 4 LGANC 192
MNO_JNL_FAILED 4 LGANC 192
MNO_JOURNALMODEL_CATLG_DEL_FAIL 4 LGANC 192
MNO_JOURNALMODEL_CATLG_FAIL 4 LGANC 192
MNO_JOURNALMODEL_DISCARDED 4 LGANC 192
MNO_JOURNALMODEL_INSTALLED 4 LGANC 192
MNO_JOURNALMODEL_REPLACED 4 LGANC 192
MNO_LOOP 1 USANC 406
MNO_LOOP 1 XSANC 449
MNO_LOOP 4 SMDCC 362
MNO_NO_MVS_STORAGE 1 USANC 406
MNO_NO_MVS_STORAGE 1 XSANC 449
MNO_NO_MVS_STORAGE 4 SMDCC 362
MNO_NO_SHUNTED_UOWS 4 RMUW 336, 340
MNO_NO_STOR_PROT 4 SMDCC 362
MNO_NO_STORAGE 1 USANC 406
MNO_NO_STORAGE 1 XSANC 449
MNO_NO_STORAGE 4 LGANC 192
MNO_NO_STORAGE 4 SMDCC 362
MNO_NO_TRAN_ISO 4 SMDCC 362
MNO_NOSTG_DFT_DSALIM 4 SMDCC 362
MNO_NOSTG_DFT_EDSALIM 4 SMDCC 362
MNO_NOSTG_DSA 4 SMDCC 362
MNO_NOSTG_REQ_DSALIM 4 SMDCC 362
MNO_NOSTG_REQ_EDSALIM 4 SMDCC 362
MNO_NOT_SOS_ABOVE 4 SMDCC 362
MNO_NOT_SOS_BELOW 4 SMDCC 362
MNO_RECON_INDOUBT_UOWS 4 RMUW 336, 340
MNO_RECON_INFLIGHT_UOWS 4 RMUW 336, 340
MNO_RECON_POST_COMMIT_UOWS 4 RMUW 336, 340
MNO_RENTPGM 4 SMDCC 362
MNO_RESYNC_CFAIL_BFAIL_UOWS 4 RMUW 336, 341
MNO_RESYNC_INDOUBT_UOWS 4 RMUW 336, 341
MNO_RESYNC_INFLIGHT_UOWS 4 RMUW 336, 341
MNO_SEVERE_ERROR 1 USANC 406
MNO_SEVERE_ERROR 1 XSANC 449
MNO_SEVERE_ERROR 4 LGANC 192
MNO_SEVERE_ERROR 4 SMDCC 362
MNO_SEVERE_ERROR 4 TSA 381
MNO_SHUNTED_UOWS 4 RMUW 336, 340
MNO_SOS_ABOVE 4 SMDCC 362
MNO_SOS_BELOW 4 SMDCC 362
MNO_STCK_ERROR 1 USANC 406
MNO_STCK_ERROR 1 XSANC 449
MNO_STCK_ERROR 4 SMDCC 362
MNO_STOR_PROT 4 SMDCC 362
MNO_STOR_PROT_REQ 4 SMDCC 362
MNO_STORAGE_VIOLATION 4 SMDCC 362
MNO_STREAM_CONN_CONFLICT 4 LGANC 192
MNO_STREAM_CONN_FAILED 4 LGANC 192
MNO_STREAM_DEFINE_BADHLQ 4 LGANC 192
MNO_STREAM_DEFINE_ERROR 4 LGANC 192
MNO_STREAM_DEFINE_INVSPACE 4 LGANC 192
MNO_STREAM_DEFINE_LIKE 4 LGANC 192
MNO_STREAM_DEFINE_MAXSTREAM 4 LGANC 192
MNO_STREAM_DEFINE_NOAUTH 4 LGANC 192
MNO_STREAM_DEFINE_NOSTRUCTNAME 4 LGANC 192
MNO_STREAM_DEFINE_STREAMNAME 4 LGANC 192
MNO_STREAM_DEFINE_STRUCTNAME 4 LGANC 192
MNO_STREAM_DEFINED 4 LGANC 192
MNO_STREAM_ENQ_CONFLICT 4 LGANC 192
MNO_SUCCESSFUL_KEYPOINT 4 RMUW 336, 341
MNO_TRAN_ISO 4 SMDCC 362
MNO_TRAN_ISO_REQ 4 SMDCC 362
MODE (7F) BAACT 17
MODE (9F) BAACT 11
MODE_ACTIVE (BIT) DSANC 57, 60
MODE_ACTIVE 1 BAACT 20
MODE_CANCELLING 1 BAACT 20
MODE_COMPLETE 1 BAACT 20
MODE_DORMANT 1 BAACT 20
MODE_INITIAL 1 BAACT 20
MODE_NAME (3C) CPCPS 32
MODE_NAME_LENGTH (38) CPCPS 32
model
  temporary storage model class, TSMN 390
MODH_AUTOREG_13 161
MODH_EYE_CATCHER 8 KEMHD 162
MODH_HANDLE_DEF_ABEND (BIT) KEMHD 161
MODH_IPROC_D (34) KEMHD 162
MODH_IPROC_F (36) KEMHD 162
MODHAM31 (BIT) KEMHD 162
MODHATNR 1 KEMHD 162
MODHATR1 (26) KEMHD 161
MODHATR2 (27) KEMHD 161
MODHATRD 1 KEMHD 162
MODHATRE 1 KEMHD 162
MODHCMS (BIT) KEMHD 161
MODHCNUM (3B) KEMHD 162
MODHDATE (18) KEMHD 161
MODHDOS (BIT) KEMHD 161
MODHEYE (2) KEMHD 161
MODHHLEN (0) KEMHD 161
MODHIPROC (34) KEMHD 162
MODHLANG (B) KEMHD 161
MODHLEVL (A) KEMHD 161
MODHMLEN 162
MODHNAME (10) KEMHD 161
MODHOS (BIT) KEMHD 161
MODHRCVR (28) KEMHD 162
MODHRELS 161
MODHSERV (2C) KEMHD 162
MODHSMODE (44) KEMHD 162
MODHSMODE_24 4 KEMHD 162
MODHSMODE_31 4 KEMHD 162
MODHSNUM (3A) KEMHD 162
MODHSOFF (38) KEMHD 162
MODHSTKL (40) KEMHD 162
MODHSYST (C) KEMHD 161
MODHTIME 161
module
  kernel module header, KEMHD 161
MODULE_DESCRIPTOR (0) KEMHD 161
monitoring
  monitoring authorised parameter block, MNAFB 260
  monitoring domain control blocks, MNCBS 262
MONITORING_INITIALISED 2 MNCBS 273
MONITORING_INITIALISING 2 MNCBS 273
MONITORING_QUIESCED 2 MNCBS 273
MONITORING_QUIESCING 2 MNCBS 273
MONITORING_TERMINATED 2 MNCBS 273
MONITORING_TERMINATING 2 MNCBS 273
MORE_TO_ANALYSE (BIT) PAA 283
MOST_RECENT_USE (78) DSANC 59
MOVE_IN_PROGRESS (BIT) L2CH 220
MOVE_IN_PROGRESS (BIT) RMLK 312
MOVE_IN_PROGRESS (BIT) RMUW 331
MSG_LEVEL (BIT) MEPS 257
MSG_LEVEL_INFO (240) MEPS 257
MSG_MOD_PTRS 257
MSG_TABLE_ADDR (908) STUCB 376
MSL_WARNING_MSG (1A8) L2BS 216
MSL_WARNING_MSG (1A8) L2SR 248
MSL_WARNING_MSG (B8) L2HS 230
MULTIPLE_TCBS (1A1) DSANC 57
MULTIPLE_TCBS (21) DSANC 60
MUST_CLOSE (BIT) XCCBC 433
MVS 1 DSTSK 67
MVS_BLOCK_HEADER 210
MVS_EXTENSION (8) DSTSK 67
MVS_STREAM_NAME (108) L2BS 215
MVS_STREAM_NAME (108) L2SR 247
MVS_STREAM_NAME (18) L2HS 230
MVS_STREAM_TOKEN (14C) L2BS 216
MVS_STREAM_TOKEN (14C) L2SR 248
MVS_STREAM_TOKEN (5C) L2HS 230
MVSLOGBLOCKHEADER (0) L2LF 231
MXT_ADJUSTMENT 4 SMDCC 356

N

N (0) BAACT 13
N (14) SOA 371
N (14) XSXD 455
N (1C) XSXD 455
N (20) XSANC 448
N (28) XSANC 448
N (2C) DHANC 39
N (34) DHANC 39

N (34) USANC 405
N (34) XSSS 451
N (38) SOA 367
N (3C) DHANC 39
N (3C) USANC 405
N (3C) XSSS 452
N (4) XSXD 455
N (40) SOA 367
N (44) DHANC 39
N (44) USANC 405
N (48) SOA 367
N (4C) DHANC 39
N (4C) USANC 405
N (50) SOA 367
N (54) DHANC 39
N (54) USANC 405
N (58) SOA 367
N (5C) DHANC 39
N (60) SOA 367
N (64) DHANC 39
N (6C) USANC 406
N (74) USANC 406
N (84) SOA 370
N (94) BAACT 17
N (B4) BAACT 11
N (C) UDB 403
N (C) XSXD 455
name
  temporary storage name class, TSNM 393
NAME (0) BAPT 23
NAME (0) PTE 299
NAME (10) PTE 298
NAME (10) RMID 303
NAME (10) RMLI 304
NAME (10) RMLK 315
NAME (10) RMNS 323
NAME (10) RMRO 325
NAME (10) XSSS 455
NAME (18) UDB 404
NAME (1C) RMDM 301
NAME (1C0) RMLK 315
NAME (1C0) RMUW 334
NAME (20) L2DM 224
NAME (3) DSTBA 63
NAME (44) UDB 404
NAME (5A) RMNM 320
NAME (78) RMUW 338
NAME (8B8) RMLK 306
NAME_ADDR (20) SOA 372
NAME_LENGTH (1C) SOA 372
NAME_PART (10) PTE 298
NDX (0) DUFC 75
NDX_BLOCK_ADDRESS (8) DUFC 75
NDX_BLOCK_LENGTH (C) DUFC 75
NDX_BLOCK_NAME (14) DUFC 75
NDX_NEXT (0) DUFC 75
NDX_NEXT2 (4) DUFC 75
NDX_PAGE_NUMBER (10) DUFC 75
NETNAME (10) PTE 299
NETNAME (28) PTE 298
NETWORK (20) PTE 298
NETWORK (8) PTE 299
NEW_SIGNAL_MASK (1C) SOA 372
NEW_STATE_AFTER_BACKOUT_RULES 33
NEW_TASK_DELAY (50) DSANC 54
NEW_TASK_MINUS (7C) DSANC 54
NEW_TASK_PENALTY (64) DSANC 54
NEXT (1C) RMLS 318
NEXT (24) BAACT 9
NEXT (24) L2BS 212, 218
NEXT (24) L2CH 220, 222
NEXT (24) L2SR 243, 249, 250
NEXT (24) RMLK 309, 311
NEXT (24) RMNS 322
NEXT (24) RMUW 330, 335
NEXT (27C) L2BS 217
NEXT (2C) RMLK 305
NEXT (2C) RMLS 318
NEXT (2C) RMNS 323
NEXT (2C) RMUW 337
NEXT (34) BAACT 9
NEXT (34) L2BS 218
NEXT (34) L2CH 222
NEXT (34) L2SR 249, 250
NEXT (34) RMLK 309
NEXT (34) RMNS 322
NEXT (3C) RMLK 305
NEXT (3C) RMNS 323
NEXT (3C) RMSL 327, 329
NEXT (3C) RMUW 337
NEXT (4C) RMSL 327, 329
NEXT (54) RMLK 316
NEXT (5C) L2BS 213
NEXT (5C) L2SR 244
NEXT (64) RMLK 316
NEXT (6C) L2BS 213
NEXT (6C) L2SR 244
NEXT (74) L2CH 223
NEXT (74) RMUW 338
NEXT (7C) BAACT 6
NEXT (7C) RMLK 312
NEXT (7C) RMUW 332
NEXT (80) BAACT 13, 19
NEXT (8B4) RMLK 306
NEXT (8C) BAACT 6
NEXT (8C) RMLK 312
NEXT (8C) RMUW 332
NEXT (92C) RMLK 307
NEXT (93C) RMLK 307
NEXT (954) RMUW 339
NEXT (964) RMUW 339
NEXT (9C) L2BS 213
NEXT (9C) L2SR 244
NEXT (AC) L2BS 213
NEXT (AC) L2SR 244
NEXT (C) BAACT 21
NEXT (C) L2CH 221
NEXT (C) L2SR 251
NEXT (C) RMID 303
NEXT (C) RMLI 304
NEXT (C) RMLK 315
NEXT (C) RMNM 321
NEXT (C) RMNS 322
NEXT (C) RMUW 334, 335
NEXT (CC) BAACT 18
NEXT (D4) RMLK 313
NEXT (D4) RMUW 333, 338
NEXT (DC) BAACT 18
NEXT (E4) RMLK 313
NEXT (E4) RMUW 333, 338
NEXT (EC) BAACT 11
NEXT (FC) BAACT 11
NEXT_BLOCK_PTR (40) L2BS 212
NEXT_BLOCK_PTR (40) L2SR 244
NEXT_BLOCK_PTR (8) L2SR 250
NEXT_BLOCK_PTR (80) L2BS 213
NEXT_BLOCK_PTR (80) L2SR 244
NEXT_CE_TIME (128) DSANC 56
NEXT_COLL_EOD (70) STCB1 375
NEXT_CPC_PTR (18) CPCPS 32
NEXT_DEAD_DS_TCB (DC) DSANC 59
NEXT_ELEM (0) BAACT 14
NEXT_FREE_SUBD (17C) DSANC 56
NEXT_ID 57, 60
NEXT_IN_BROWSE (78) L2CH 221
NEXT_LL_CONCATENATED 33
NEXT_OPEN_FREE (88) DSANC 59
NEXT_RECOVERY_STATUS (9F3) RMLK 308
NEXT_RECOVERY_STATUS (EB) RMLK 311
NEXT_SHP_TIME (130) DSANC 56
NEXT_SINGLE_UPDATER (9F4) RMLK 308
NEXT_SINGLE_UPDATER (EC) RMLK 311
NEXT_TCB (10) DSANC 58
NEXT_TCP_DISPATCH_TIME (150) DSANC 56
NEXT_TI_EVENT (140) DSANC 56
NEXT_TIMEOUT_TIME (138) DSANC 56
NLS_CODE (0) MEPS 258
NLS_SUFFIX (3) MEPS 258
NLS_TABLE (0) MEPS 258
NLS_TABLE_PTR 257
NO 0 MEPS 259
NO 0 PAA 284
NO 0 TIA 380
NO_BROWSE_IN_PROGRESS 4 L2BS 219
NO_COLLECTIONS (64) STCB1 375
NO_DATA 4 L2HS 231

NO_JOURNAL 251
NO_MORE_DATA 4 BAPT 24
NO_RESYNC_OUTCOME (83) RMLK 317
NO_RESYNC_OUTCOME (9DF) RMLK 308
NO_RESYNC_OUTCOME (D7) RMLK 310
NO_SMF_WRITES (68) STCB1 375
NO_SOURCE 251
node
    node descriptor, FEP10 132
NODE (0) DDBSC 35
NODE0 (20) RMLS 318
NODE0 (28) BAACT 9
NODE0 (28) L2BS 218
NODE0 (28) L2CH 222
NODE0 (28) L2SR 249, 250
NODE0 (28) RMNS 322
NODE0 (30) RMLK 305
NODE0 (30) RMNS 323
NODE0 (30) RMUW 337
NODE0 (40) RMSL 327, 329
NODE0 (58) RMLK 316
NODE0 (60) L2BS 213
NODE0 (60) L2SR 244
NODE0 (80) BAACT 6
NODE0 (80) RMLK 312
NODE0 (80) RMUW 332
NODE0 (958) RMUW 339
NODE0 (A0) L2BS 213
NODE0 (A0) L2SR 244
NODE0 (D0) BAACT 18
NODE0 (D8) RMLK 313
NODE0 (D8) RMUW 333, 338
NODE0 (F0) BAACT 11
NOEL (18) DDBSC 35
NON_MOVED_CHAIN_HEADER (0) L2LF 235
NON_MOVED_RM_START (14) L2LF 235
NON_MOVED_RM_START (34) L2LF 236
NON_MOVED_RM_START (34) LGSF 201
NON_SYSTEM 1 DSTSK 67
NORMAL_CHAIN_HEADER (0) L2LF 233
NORMAL_RM_START (14) L2LF 234
NORMAL_RM_START (34) L2LF 235
NORMAL_RM_START (34) LGSF 200
NOSEQ_WRITE_NUMBER (A54) CCGD 30
NOT_DISABLED 4 BAPT 24
NOT_FOUND 1 LDCBS 176
NOT_POSSIBLE 251
 NQA 275
NQA (0) NQA 275
NQA_CHAIN_POINTERS (10) NQA 275
NQA_DEFAULT_INTERPRETER (60) NQA 275
NQA_DISPATCHER_POOL (68) NQA 275
NQA_DOMAIN_LOCK (3C) NQA 275
NQA_END (70) NQA 275
NQA_EYECATCHER (2) NQA 275
NQA_FIRST_BROWSE (14) NQA 275
NQA_FIRST_POOL (10) NQA 275
NQA_FLAGS (59) NQA 275
NQA_GENERAL_SUBPOOL (1C) NQA 275
NQA_INITIALISED 1 NQA 276
NQA_INITIALISING 1 NQA 276
NQA_LAST_RESET_TIME (50) NQA 275
NQA_LENGTH (0) NQA 275
NQA_LOCKS (3C) NQA 275
NQA_MISCELLANEOUS (58) NQA 275
NQA_NQEA_SUBPOOL (2C) NQA 275
NQA_NQPL_SUBPOOL (24) NQA 275
NQA_NQRN_DIRECTORY (64) NQA 275
NQA_NQRN_SUBPOOL (34) NQA 275
NQA_NQRNAME_LIST (18) NQA 275
NQA_NQRNAME_LOCK (40) NQA 275
NQA_NUM_ENQUEUE_POOLS (5C) NQA 275
NQA_PREFIX (0) NQA 275
NQA_QUIESCED 1 NQA 276
NQA_QUIESCING 1 NQA 276
NQA_STATE (58) NQA 275
NQA_STATISTICS (48) NQA 275
NQA_STATS_BUFFER_LEN (4C) NQA 275
NQA_STATS_BUFFER_PTR (48) NQA 275
NQA_SUBPOOLS (1C) NQA 275
NQA_TERMINATED 1 NQA 276
NQA_TERMINATING 1 NQA 276
NQA_XRSINDI_ACTIVE (BIT) NQA 275
 NQB 276
NQB (0) NQB 276
NQB_BROWSING_TRANID (44) NQB 276
NQB_BROWSING_TRANNUM (48) NQB 276
NQB_BROWSING_TXN_TOKEN (4C) NQB 276
NQB_CURRENT_ENQUEUE_OWNER (3C) NQB 276
NQB_CURRENT_UOW_TOKEN (30) NQB 276
NQB_CURRENT_UOWID (28) NQB 276
NQB_ENQSCOPE (BIT) NQB 276
NQB_EYECATCHER (2) NQB 276
NQB_FLAGS (18) NQB 276
NQB_LENGTH (0) NQB 276
NQB_NAME_FILTER (58) NQB 276
NQB_NAME_LENGTH 276
NQB_NEXT_BROWSE_ELEMENT (10) NQB 276
NQB_OWNER_EXTENSION (34) NQB 276
NQB_PREFIX (0) NQB 276
NQB_RMWT_BROWSE_TOKEN (14) NQB 276
NQB_SCOPE_FILTER (1C) NQB 276
NQB_STABLE_ENQUEUES (BIT) NQB 276
NQB_STABLE_NQEA (40) NQB 276
NQB_UOWID_FILTER (20) NQB 276
NQB_WAITER_EXTENSION (38) NQB 276
 NQEA 277
NQEA (0) NQEA 277
NQEA_ACTIVE_START_TIME 278
NQEA_CLEARED_FIELDS (10) NQEA 277
NQEA_CLEARED_FLAGS1 (14) NQEA 277
NQEA_CLEARED_FLAGS2 (15) NQEA 277
NQEA_ENQSCOPE (50) NQEA 278
NQEA_EYECATCHER (0) NQEA 277
NQEA_FIXED_LENGTH 4 NQEA 278
NQEA_HASH_NEXT (C) NQEA 277
NQEA_HASH_PREV (8) NQEA 277
NQEA_HASH_VALUE (2C) NQEA 277
NQEA_HASHMARK (58) NQEA 278
NQEA_LOCKED_FAILURES (38) NQEA 278
NQEA_LONG_NAME (BIT) NQEA 277
NQEA_MVS_GETMAINED (BIT) NQEA 278
NQEA_NAME (5C) NQEA 278
NQEA_NAME_LENGTH (58) NQEA 278
NQEA_NAME2_LENGTH (4C) NQEA 278
NQEA_NAME2_SUPPLIED (BIT) NQEA 277
NQEA_NEXT_FREE (4) NQEA 277
NQEA_NEXT_WAITER (10) NQEA 277
NQEA_NQRMODEL_POINTER (18) NQEA 277
NQEA_OWNER 277
NQEA_OWNER_SHUNTED (BIT) NQEA 277
NQEA_PERMANENT_FLAGS (35) NQEA 278
NQEA_POOL_POINTER (48) NQEA 278
NQEA_PREFIX (0) NQEA 277
NQEA_QUICKCELLABLE (BIT) NQEA 278
NQEA_RESUME_FOR_LOCKED (BIT) NQEA 277
NQEA_RESUME_REQUIRED (BIT) NQEA 277
NQEA_RETAINED (BIT) NQEA 277
NQEA_RETAINED_START_TIME (40) NQEA 278
NQEA_SHUNT_ACTION_OVERRIDE (34) NQEA 278
NQEA_SHUNT_OVERRIDE (BIT) NQEA 277
NQEA_SHUNTED_OWNER (24) NQEA 277
NQEA_SUSPEND_TOKEN (30) NQEA 278
NQEA_SYSENQ_ECB (54) NQEA 278
NQEA_SYSENQ_GRANTED (BIT) NQEA 277
NQEA_SYSENQ_WAITING (BIT) NQEA 277
NQEA_SYSPLEX_SCOPE (BIT) NQEA 277
NQEA_TRANSACTION_COUNT (1C) NQEA 277
NQEA_UOW_COUNT (20) NQEA 277
NQEA_UOW_NEXT 277
NQEA_WAIT_START_TIME (40) NQEA 278
NQEA_WAITER (BIT) NQEA 277
 NQOX 279
NQOX (0) NQOX 279
NQOX_DEFAULT_MAX_SLOTS 4 NQOX 279
NQOX_ENQUEUE_NAME_LEN (30) NQOX 279
NQOX_ENQUEUE_NAME_PTR (34) NQOX 279
NQOX_ENQUEUE_OWNER (28) NQOX 279
NQOX_ENQUEUE_POOL (2C) NQOX 279
NQOX_EYECATCHER (4) NQOX 279
NQOX_LENGTH (0) NQOX 279
NQOX_MAXIMUM_SLOTS (18) NQOX 279
NQOX_OWNER_SLOT 279
NQOX_PERM_SLOTS_USED (20) NQOX 279
NQOX_PREFIX (0) NQOX 279
NQOX_SPARE_NAME_STG_LEN (14) NQOX 279

NQOX_SPARE_NAME_STG_PTR (10) NQOX 279
NQOX_TEMP_SLOTS_USED (1C) NQOX 279
NQPL 280
NQPL (0) NQPL 280
NQPL_DEFAULT_INTERPRETATION 1 NQPL 281
NQPL_DEFAULT_SHUNT_ACTION (144) NQPL 280
NQPL_DEFAULT_TYPE (151) NQPL 280
NQPL_DISPATCHER_TASK (BIT) NQPL 280
NQPL_DOMAIN_LOCK_COPY (C) NQPL 280
NQPL_END (180) NQPL 281
NQPL_ENQUEUE_INTERPRETATION 280
NQPL_ERROR_LEVEL (145) NQPL 280
NQPL_EXEC_INTERPRETER (150) NQPL 280
NQPL_EYECATCHER (0) NQPL 280
NQPL_FIRST_CDS_COUNT (10) NQPL 280
NQPL_FIRST_FREE_NQEA (14) NQPL 280
NQPL_FLAGS1 (146) NQPL 280
NQPL_FREE_NQEA_CHAIN (10) NQPL 280
NQPL_GLOBAL_WAITED (174) NQPL 281
NQPL_GLOBAL_WAITED_TIME (178) NQPL 281
NQPL_HASH_CONSTANT (20) NQPL 280
NQPL_HASH_CONSTANT_VALUE 4 NQPL 281
NQPL_HASH_MASK (1C) NQPL 280
NQPL_HASH_MASK_VALUE 4 NQPL 281
NQPL_HASH_TABLE (40) NQPL 280
NQPL_HASHSIZE 4 NQPL 281
NQPL_HASHSIZE_MINUS_1 4 NQPL 281
NQPL_INTERPRETER_ADDR 281
NQPL_MISCELLANEOUS (144) NQPL 280
NQPL_NEXT_POOL (140) NQPL 280
NQPL_NO_INTERPRETATION 1 NQPL 281
NQPL_OWN_INTERPRETER 1 NQPL 281
NQPL_POOL_NAME (4) NQPL 280
NQPL_PREFIX (0) NQPL 280
NQPL_QUICKCELL_NAME_LENGTH (18) NQPL 280
NQPL_RETURN_EXCEPTION 1 NQPL 281
NQPL_RETURN_INVALID 1 NQPL 281
NQPL_SECTION_1 (0) NQPL 280
NQPL_SECTION_2 280
NQPL_SECTION_3 (140) NQPL 280
NQPL_STATISTICS_1 (24) NQPL 280
NQPL_STATISTICS_2 (158) NQPL 281
NQPL_SYSPLEX_SCOPE (BIT) NQPL 280
NQPL_TOTAL_BUSY (28) NQPL 280
NQPL_TOTAL_LOCKED_IMMED (158) NQPL 281
NQPL_TOTAL_LOCKED_WAITED (15C) NQPL 281
NQPL_TOTAL_PURGED_CANCELLED (160) NQPL 281
NQPL_TOTAL_PURGED_TIMED_OUT (164) NQPL 281
NQPL_TOTAL_REQUESTS (24) NQPL 280
NQPL_TOTAL_RETAINED (168) NQPL 281
NQPL_TOTAL_RETAINED_TIME (16C) NQPL 281
NQPL_TOTAL_WAITED (2C) NQPL 280
NQPL_TOTAL_WAITED_TIME (30) NQPL 280
NQPL_TYPE_DATASET 1 NQPL 281
NQPL_TYPE_DISPATCHER 1 NQPL 281
NQPL_TYPE_EXECENQ 1 NQPL 281
NQPL_TYPE_EXECENQADDR 1 NQPL 281
NQPL_TYPE_EXECENQPLEX 1 NQPL 281
NQPL_TYPE_FILE 1 NQPL 281
NQPL_TYPE_TDQUEUE 1 NQPL 281
NQPL_TYPE_TSQUEUE 1 NQPL 281
NQWX 282
NQWX (0) NQWX 282
NQWX_DEFAULT_MAX_SLOTS 4 NQWX 282
NQWX_ENQUEUE_WAITER (18) NQWX 282
NQWX_EYECATCHER (4) NQWX 282
NQWX_LENGTH (0) NQWX 282
NQWX_MAXIMUM_SLOTS (10) NQWX 282
NQWX_PREFIX (0) NQWX 282
NQWX_SLOTS_USED (14) NQWX 282
NQWX_WAITER_SLOT (18) NQWX 282
NUCLEUS_POOLS_BDY 2 LDCBS 175
NUCLEUS24_POOL 4 LDCBS 174
NUCLEUS24_POOL_NAME 8 LDCBS 174
NUCLEUS24_RESIDENT_POOL 4 LDCBS 174
NUCLEUS24_RESIDENT_POOL_NAME 8 LDCBS 174
NUCLEUS24_RESIDENT_RO_POOL 4 LDCBS 174
NUCLEUS24_RESIDENT_RO_POOL_NAME 8 LDCBS 175
NUCLEUS24_RO_POOL 4 LDCBS 174
NUCLEUS24_RO_POOL_NAME 8 LDCBS 174
NUCLEUS31_POOL 4 LDCBS 174
NUCLEUS31_POOL_NAME 8 LDCBS 174
NUCLEUS31_RESIDENT_POOL 4 LDCBS 174
NUCLEUS31_RESIDENT_POOL_NAME 8 LDCBS 175
NUCLEUS31_RESIDENT_RO_POOL 4 LDCBS 174
NUCLEUS31_RESIDENT_RO_POOL_NAME 8 LDCBS 175
NUCLEUS31_RO_POOL 4 LDCBS 174
NUCLEUS31_RO_POOL_NAME 8 LDCBS 174
NULL_LANGUAGE 1 MEPS 259
NULL_LOGSTREAM_TOKEN 4 L2SL 241
NULL_PRO_REF (0) BAACT 7
NULL_RMRO_FORCE_TOKEN 4 RMRO 326
NULL_SYSTEM_LOG_CHAIN_TOKEN 4 RMUW 336, 340
NULL_UOW_BROWSE_TOKEN 4 RMUW 336, 340
NULL_UOW_TOKEN 4 RMUW 336, 340
NUM_APPLID_IGNORE (80C) STUCB 375
NUM_APPLID_SELECT 375
NUM_OPEN_TYPES 1 SMDCC 363
NUM_OPEN_TYPES 1 XMXDC 444
NUM_OWNED_OPEN_TCBS (C8) DSTSK 66
NUM_SUBSPACE_OPEN_TYPES 1 SMDCC 363
NUM_SUBSPACE_OPEN_TYPES 1 XMXDC 444
NUM_TASKS 54
NUM_THREADS (34) CCGD 29
NUMBER (BIT) L2BL 208
NUMBER_MSGSFDS (18) SOA 371
NUMBER_OF_BLOCKS (110) RMUW 339
NUMBER_OF_BLOCKS (470) RMLK 306
NUMBER_OF_BLOCKS (50) RMLK 305
NUMBER_OF_BLOCKS (530) RMUW 339
NUMBER_OF_ENF_EVENTS 4 DMENC 53
NUMBER_OF_LANGS (1C) MEPS 257
NUMBER_OF_LANGUAGE_CODES 2 MEPS 259
NUMBER_OF_SUBTASKS (10) DSANC 54

O

object
  recovery manager loggable object identity instance, RMLI 304
OBJECT_FACTORY (10) L2BL 209
OBJECT_TOKEN (0) L2LT 238
OF_EYE_CATCHER (10) BAACT 17
OF_EYE_CATCHER (10) L2BL 209
OF_EYE_CATCHER (38) L2BS 219
OF_EYE_CATCHER (38) L2CH 222
OF_EYE_CATCHER (38) L2SR 249
OF_EYE_CATCHER (40) RMUW 337
OF_EYE_CATCHER (880) RMLK 306
OFF 0 MEPS 259
OFF 0 PAA 284
OFF 0 TIA 380
OK 1 CCGD 31
OLD_SIGNAL_MASK (24) SOA 372
OLDC 1 DSTSK 67
OLDW 1 DSTSK 67
ON 0 MEPS 259
ON 0 PAA 284
ON 0 TIA 380
OP_ID (1C) RMUW 336
OP_ID (4F) RMLK 311
OP_ID (4F) RMUW 331
OPEN (BIT) XCCBC 433
OPEN 0 PAA 284
OPEN_DS_TCB 66
OPEN_DS_TCB_STATE 59
OPEN_FLAGS (660) DSANC 58
OPEN_FLAGS (8C) DSANC 59
OPEN_FLAGS (C0) DSTSK 66
OPEN_INDEX (1AC) DSANC 57
OPEN_INDEX (2C) DSANC 60
OPEN_MODE (BIT) DSANC 57, 59, 60
OPEN_PLIST_A (24) CCGD 29
OPEN_SECONDARY (BIT) L2SL 241
OPEN_STATUS (2F) CCGD 29
OPEN_TCB_MANAGEMENT_LOCK (644) DSANC 58
OPEN_TCBS (644) DSANC 58
OPEN_TCBS (B8) DSTSK 66
OPENING_SYSIN (BIT) PAA 283
OPERATION (1C) SOA 372
OPTIMAL_CLIENTS_ONLY (BIT) RMLK 314
OPTIMAL_CLIENTS_ONLY (BIT) RMUW 333
OPTION_BLOCK 8 LDCBS 175
OPTION_DATA_ADDR (2C) SOA 372
OPTION_DATA_LENGTH (28) SOA 372
OPTION_NAME (24) SOA 372

ORIGIN (70) BAACT 15
ORIGIN_TRANID (107) BAACT 16
OTHER_SWITCHES (901) STUCB 376
OUT_OF_RANGE 251
OUT_OF_RANGE 4 L2SL 241
OUTBOUND_RECOVERY_IN_PROGRESS (BIT) RMLK 307, 310
OUTSTANDING_LL_COUNT (C8) CPCPS 33
OVERRIDE_STORE_H (20) PAA 283
OVERRIDE_STORE_L (24) PAA 283
OWN_PROCESS (0) BAACT 17
OWN_PROCESS (20) BAACT 10
OWN_ROOT_ID (64) BAACT 17
OWN_ROOT_ID (84) BAACT 11
OWNED_BY_LINKSET 307, 310
owner
    enqueue domain browse owner extension, NQOX 279
    recovery manager resource owner instance, RMRO 324
OWNER (14) L2SR 251
ownership
    temporary storage ownership lock class, TSOL 394
OWNING_TASK (84) DSANC 59

P

P (0) XSXD 455
P (10) SOA 371
P (10) XSXD 455
P (18) XSXD 455
P (1C) XSANC 448
P (24) XSANC 448
P (28) DHANC 39
P (30) DHANC 39
P (30) USANC 405
P (30) XSSS 451
P (34) SOA 367
P (38) DHANC 39
P (38) USANC 405
P (38) XSSS 451
P (3C) SOA 367
P (40) DHANC 39
P (40) USANC 405
P (44) SOA 367
P (48) DHANC 39
P (48) USANC 405
P (4C) SOA 367
P (50) DHANC 39
P (50) USANC 405
P (54) SOA 367
P (58) DHANC 39
P (5C) SOA 367
P (60) DHANC 39
P (68) USANC 406
P (70) USANC 406
P (8) UDB 403
P (8) XSXD 455
P (80) SOA 370
PA_CATALOG_SUFFIX (0) PAA 284
PA_RECORD_TYPE (2) PAA 284
PAA 283
PAA_ARROW (2) PAA 283
PAA_BLOCK_NAME (8) PAA 283
PAA_DFH (3) PAA 283
PAA_DM_FLAGS (10) PAA 283
PAA_DOMID (6) PAA 283
PAA_IO_FLAGS 283
PAA_LENGTH (0) PAA 283
PAA_MORE_IO_FLAGS (12) PAA 283
PAA_PREFIX (0) PAA 283
PADM_ERROR_RECOVERY (BIT) PAA 283
PADM_NAME 7 PAA 284
PAGE_NUMBER (818) STUCB 375
PAGE_SIZE 2 PAA 284
PAGEROUND 4 SMDCC 362
PAGESIZE (814) STUCB 375
PAGESIZE 4 SMDCC 362
PAGP_NAME 7 PAA 284
PAIO_NAME 7 PAA 284
PAM_ADDR (B0) DSANC 55
PAM_ADDR (C0) DSANC 55
PAM_ADDR (D0) DSANC 55
PAM_ADDR (E0) DSANC 55
PAM_ADDR (F0) DSANC 55
PAPA (8) DDBSC 35
parameter
    monitoring authorised parameter block, MNAFB 260
    parameter area declarations, DUFP 76
    parameter manager domain anchor block, PAA 283
    request parameter area, FEP17 141
    statistics authorised parameter block, STAFB 373
parameters
    web business logic interface parameters, WBBLC 416
PARENT_ADD (7C) BAACT 13, 19
PARENT_GENERATION 11, 17
PARENT_KEY (32) BAACT 17
PARENT_KEY (52) BAACT 10
PARENT_MODENAME (1AA) DSANC 57
PARENT_MODENAME (2A) DSANC 60
PARENT_TRANID (80) BAACT 17
PARENT_TRANID (A0) BAACT 11
PARENT_USERID (84) BAACT 17
PARENT_USERID (A4) BAACT 11
PARM_SAVE_AREA (0) PAA 284
PARM_SAVE_AREA_P (1C) PAA 283
PARM_SAVE_AREA_SIZE (0) PAA 284
PARM_SAVE_ARROW (2) PAA 284
PARM_SAVE_BLOCK_NAME (8) PAA 284
PARM_SAVE_DFH (3) PAA 284
PARM_SAVE_DOMID (6) PAA 284
PARM_SAVE_PREFIX (0) PAA 284
parms
    web error program parms, WBEPC 419
PARMS (12) PAA 284
PARMS_LEN (10) PAA 284
PARTIAL_ID 33
PARTIAL_ID_RECEIVED (BIT) CPCPS 33
partner
    partner domain static storage area, PRS 296
    partner table entry, PTE 297
PARTNER_LU_NAME (48) CPCPS 33
PARTNER_LU_NAME_LENGTH (44) CPCPS 33
PASS (9EC) RMLK 308
PASS (E4) RMLK 311
PASS_AKP (BIT) L2SL 241
PASY_EP_PTR (30) PAA 283
PATCH_SPACE (933) STUCB 377
PBB 400
PBB_NEXT (0) TSRL 400
PBB_POOL_NAME (18) TSRL 400
PBB_PREFIX (0) TSRL 400
PBB_PREV (4) TSRL 400
PBB_TRANID (8) TSRL 400
PBB_TRANNUM (C) TSRL 400
PBB_TRANTOKEN (10) TSRL 400
PCA (0) TSRL 399
PCA_CONNECT_FAILED (BIT) TSRL 400
PCA_CONNECT_TOKEN (18) TSRL 400
PCA_FLAGS (1C) TSRL 400
PCA_NEXT (0) TSRL 400
PCA_POOL_NAME (8) TSRL 400
PCA_PREFIX (0) TSRL 399
PCA_PREV (4) TSRL 400
PCA_WAIT_QUEUE (10) TSRL 400
PCHAIN (38) RMNS 323
PCHAINNODE 321
PDB (0) LDCBS 173
PDB_CATALOG_MODULE (5) LDCBS 173
PDB_CREATION_STCK 173
PDB_DESCRIPTOR_FIELDS (0) LDCBS 173
PDB_EXECUTION_KEY (6) LDCBS 173
PDB_PROGRAM_ATTRIBUTE (2) LDCBS 173
PDB_PROGRAM_TYPE (0) LDCBS 173
PDB_PROGRAM_USAGE (1) LDCBS 173
PDB_REQUIRED_AMODE (4) LDCBS 173
PDB_REQUIRED_RMODE (3) LDCBS 173
PEAK_NUM_TASKS (70) DSANC 54
PERFORM_AFTER_WAIT_UEXIT 54
PERFORM_BEFORE_WAIT_UEXIT 54
PERFORM_KE_READ_TIME (BIT) DSANC 59
PERMANENT_PTR (0) BAACT 6, 7
PERMANENT_PTR (74) BAACT 12, 19
PERMANENT_STATE (20) BAACT 10
PERSIST (0) WRB 430
PERSIST_NO 1 WRB 430
PERSIST_YES 1 WRB 430
PERSISTENT_DATA (10) RMNM 321

PERSISTENT_NAME (18) RMLK 316
PERSISTENT_NAME (38) RMNS 323
PERSISTENT_NAME (88) RMNM 321
PESA (0) PGA 285
PESA_AMODE (B) PGA 285
PESA_ARROW (2) PGA 285
PESA_BLOCK_NAME (6) PGA 285
PESA_CALEN (24) PGA 285
PESA_COMMON_CONTROL_AREA (1E0) PGA 285
PESA_DFH (3) PGA 285
PESA_EDF_REPLY (204) PGA 285
PESA_EIS_APLI_SAVEAREA (10) PGA 285
PESA_EIS_EXEC_DATA (26) PGA 285
PESA_EIS_SUPERLINK_DATA (F0) PGA 285
PESA_EISTG (20) PGA 285
PESA_EIUS_EXEC_DATA (5A) PGA 285
PESA_EIUS_SUPERLINK_STACK (175) PGA 285
PESA_END 285
PESA_ENVIRONMENT_TYPE (A) PGA 285
PESA_EXEC 4 PGA 286
PESA_EXEC_SPECIFIC (20) PGA 285
PESA_EXEC_SPECIFIC_END 285
PESA_FLAG2 (205) PGA 285
PESA_FLAG3 (206) PGA 285
PESA_FLAG5 (207) PGA 285
PESA_FLAGS (205) PGA 285
PESA_GLUE 4 PGA 286
PESA_GLUE_SPECIFIC (1E0) PGA 285
PESA_LENGTH (0) PGA 285
PESA_LENGTH_EXEC 4 PGA 286
PESA_LENGTH_GLUE 4 PGA 286
PESA_LENGTH_PLT 4 PGA 286
PESA_LENGTH_SYSTEM 4 PGA 286
PESA_LENGTH_TRUE 4 PGA 286
PESA_LENGTH_URM 4 PGA 286
PESA_PCTWA (14) PGA 285
PESA_PLT 4 PGA 286
PESA_PREFIX (0) PGA 285
PESA_PREV (C) PGA 285
PESA_STANDARD (0) PGA 285
PESA_STANDARD_END (20) PGA 285
PESA_SUPERLINK_SPECIFIC (F0) PGA 285
PESA_SUPERLINK_SPECIFIC_END 285
PESA_SYSTEM 4 PGA 286
PESA_SYSTEM_EIB (120) PGA 285
PESA_TCAEISFL (1DA) PGA 285
PESA_TRUE 4 PGA 286
PESA_URM 4 PGA 286
PESA_USER_EIB (185) PGA 285
PEX_NUM (64) DSANC 61
PG_TRANSACTION_TOKEN (0) PGHM 294
PGA 285
PGA_ARROW (2) PGDCC 286
PGA_ATTEMPTED_AUTOINSTALLS (70) PGDCC 287
PGA_AUTOINSTALL_CATALOG_STATE (64) PGDCC 286
PGA_AUTOINSTALL_EXIT_NAME (68) PGDCC 286
PGA_AUTOINSTALL_STATE (60) PGDCC 286
PGA_BLOCK_NAME (8) PGDCC 286
PGA_CATALOG_ALL 4 PGDCC 291
PGA_CATALOG_MODIFY 4 PGDCC 291
PGA_CATALOG_NONE 4 PGDCC 291
PGA_COLD_START (BIT) PGDCC 287
PGA_DFH (3) PGDCC 286
PGA_DISABLED 4 PGDCC 291
PGA_DOMID (6) PGDCC 286
PGA_ENABLED 4 PGDCC 291
PGA_FAILED_AUTOINSTALLS (78) PGDCC 287
PGA_GENERAL_SUBPOOL_TOKEN (10) PGDCC 286
PGA_HMRSA_SUBPOOL_TOKEN (40) PGDCC 286
PGA_HTB_SUBPOOL_TOKEN (38) PGDCC 286
PGA_INDICATORS (9C) PGDCC 287
PGA_INITIALISED 4 PGDCC 291
PGA_INITIALISING 4 PGDCC 291
PGA_JVMCLASS_SUBPOOL_TOKEN (20) PGDCC 286
PGA_LANGUAGES_AVAILABLE (BIT) PGDCC 287
PGA_LAST_RESET_TIME (50) PGDCC 286
PGA_LENGTH (0) PGDCC 286
PGA_LLE_SUBPOOL_TOKEN (28) PGDCC 286
PGA_LOCAL_SYSTEM_NAME (A0) PGDCC 287
PGA_LOCK_TOKEN (58) PGDCC 286
PGA_PG_AVAILABLE (BIT) PGDCC 287
PGA_PG_STATE (5C) PGDCC 286
PGA_PGWE_HEAD (8C) PGDCC 287
PGA_PGWE_SUBPOOL_TOKEN (30) PGDCC 286
PGA_PPT_DIRECTORY (7C) PGDCC 287
PGA_PPT_RECOVERY_COMPLETE (BIT) PGDCC 287
PGA_PPT_VERSION_NUMBER (80) PGDCC 287
PGA_PPTE_SUBPOOL_TOKEN (18) PGDCC 286
PGA_PREFIX (0) PGDCC 286
PGA_PTA_SUBPOOL_TOKEN (48) PGDCC 286
PGA_QUIESCED 4 PGDCC 291
PGA_QUIESCING 4 PGDCC 291
PGA_REJECTED_AUTOINSTALLS (74) PGDCC 287
PGA_SM_ACCESS_TOKEN (94) PGDCC 287
PGA_SM_ISOLATION_TOKEN (98) PGDCC 287
PGA_STORAGE_PROTECT (BIT) PGDCC 287
PGA_SYS_LLE_HEAD (84) PGDCC 287
PGA_TERMINATED 4 PGDCC 291
PGA_TERMINATING 4 PGDCC 291
PGA_XRSINDI_ACTIVE (BIT) PGDCC 287
PGANCHOR (0) PGDCC 286
PGDCC 286
PGHM 293
PGWE (0) PGDCC 290
PGWE_NEXT (0) PGDCC 290
PGWE_PPTE_PTR (C) PGDCC 290
PGWE_PREFIX (0) PGDCC 290
PGWE_PREV (4) PGDCC 290
PGWE_PROGRAM_NAME (10) PGDCC 290
PGWE_SUSPEND_TOKEN (8) PGDCC 290
PHASE_INFO (14) MEPS 257
PHASE_MANAGEMENT (10) DMCB1 47
PHS1_EXPIRY_TIME (58) DSTSK 65
PHS1_PERIOD_LENGTH (30) DSANC 54
PHS1_PRIORITY 56
PHS1_PRIORITY_BONUS (38) DSANC 54
PHS1_PRIORITY_HIGH (170) DSANC 56
PHS1_PRIORITY_LOW (174) DSANC 56
PLCB (0) PGDCC 289
PLCB_AMODE_31 289
PLCB_ANY_DATA_LOC 290
PLCB_ARROW (2) PGDCC 289
PLCB_BLOCK_NAME (8) PGDCC 289
PLCB_CA_COPY (BIT) PGDCC 290
PLCB_CA_CURRENT (38) PGDCC 290
PLCB_CA_CURRENT_LEN (3C) PGDCC 290
PLCB_CA_CURRENT_X (BIT) PGDCC 290
PLCB_CA_FLAGS (48) PGDCC 290
PLCB_CA_LINK (40) PGDCC 290
PLCB_CA_LINK_COPY (BIT) PGDCC 290
PLCB_CA_LINK_LEN (44) PGDCC 290
PLCB_CA_READONLY (BIT) PGDCC 290
PLCB_CA_STORAGE_CLASS 290
PLCB_CEDF_STATUS (BIT) PGDCC 290
PLCB_COMMAREA_INFO (38) PGDCC 290
PLCB_DFH (3) PGDCC 289
PLCB_DOMID (6) PGDCC 289
PLCB_DPLSUBSET 290
PLCB_DYNAMIC_STATUS 290
PLCB_ENTRY_POINT (24) PGDCC 289
PLCB_ENVIRONMENT (31) PGDCC 290
PLCB_ENVIRONMENT_TYPE (31) PGDCC 290
PLCB_EXEC 4 PGDCC 292
PLCB_EXIT_NUMBER (54) PGDCC 290
PLCB_FLAGS (56) PGDCC 290
PLCB_GLUE 4 PGDCC 292
PLCB_HANDLE_ABEND_PGM (BIT) PGDCC 290
PLCB_HANDLE_LEVEL_TKN 290
PLCB_HPJ_PROGRAM (BIT) PGDCC 290
PLCB_INPUTMSG_SUPPLIED (BIT) PGDCC 290
PLCB_INSTANCE_FLAGS (30) PGDCC 290
PLCB_INVOKING_PROG 290
PLCB_LANGUAGE_TOKEN (2C) PGDCC 290
PLCB_LENGTH (0) PGDCC 289
PLCB_LOAD_POINT (20) PGDCC 289
PLCB_PLT 4 PGDCC 292
PLCB_PREFIX (0) PGDCC 289
PLCB_PREV (10) PGDCC 289
PLCB_PROG_PPTE (1C) PGDCC 289
PLCB_PROGRAM_DETAILS (20) PGDCC 289
PLCB_PROGRAM_INSTANCE (14) PGDCC 289
PLCB_PROGRAM_LENGTH (28) PGDCC 290
PLCB_PROGRAM_NAME (14) PGDCC 289
PLCB_SYSEIB_REQUEST (BIT) PGDCC 290
PLCB_SYSTEM 4 PGDCC 292
PLCB_TRUE 4 PGDCC 292

PLCB_URM 4 PGDCC 292
PLCB_XCTL_IN_PROGRESS (BIT) PGDCC 290
PM_ACT_PHASE (2E) DMCB1 47
PM_ACTIVE 47
PM_ARROW (12) DMCB1 47
PM_BLOCK_NAME (18) DMCB1 47
PM_DFH (13) DMCB1 47
PM_DOM_TABLE 47
PM_DOMAIN_ID (2C) DMCB1 47
PM_DOMAIN_TOKEN (28) DMCB1 47
PM_DOMID (16) DMCB1 47
PM_LENGTH (10) DMCB1 47
PM_NO_ACTIVE_DOMAINS (24) DMCB1 47
PM_PHASE_STATE 47
PM_PREFIX (10) DMCB1 47
PM_TIME_INITIALISED (44) DMCB1 47
PM_TIME_QUIESCED (54) DMCB1 47
PM_TIME_STARTED_TO_INIT (3C) DMCB1 47
PM_TIME_STARTED_TO_QUIESCE (4C) DMCB1 47
PM_TOTAL_TIME_IN_QUEUE (34) DMCB1 47
PNAME (0) BAACT 12
point
  log manager history point class, L2HP 226
POINT_ID_LENGTH 4 MEPS 259
POLLER 312, 332
pool
  enqueue domain enqueue pool, NQPL 280
  file control cfdt pool element, FCPEC 101
  file control cfdt pool wait element, FCPWC 102
  file control cfdt uow pool block, FCUPC 107
  pool descriptor, FEP11 134
POOLNAME (0) TSMN 391
POST_BYTE (18) SOA 366
POST_BYTE (1C) SOA 368
POST_BYTE (3C) SOA 370
POST_BYTE (40) SOA 370
POST_BYTE (48) SOA 368
POST_BYTE (74) SOA 367
POST_EXIT_ADDRESS (604) DSANC 58
POST_EXIT_ENABLED (BIT) DSANC 54
POST_KEYPOINT (34) RMLI 304
POST_KEYPOINT (8DC) RMLK 307
POST_KEYPOINT (9C) RMUW 338
PPA (0) SMDCC 347
PPA_ACCESS 348
PPA_ADD_SUBPOOLS (84) SMDCC 348
PPA_ANY (BIT) SMDCC 348
PPA_ARROW (2) SMDCC 347
PPA_BLOCK_NAME (8) SMDCC 347
PPA_BOUNDARY (3C) SMDCC 348
PPA_CUSHION_RELEASED (BIT) SMDCC 348
PPA_CUSHION_RELEASES (90) SMDCC 348
PPA_CUSHION_SIZE (44) SMDCC 348
PPA_DELETE_SUBPOOLS (88) SMDCC 348
PPA_DFH (3) SMDCC 347
PPA_DOMAIN_FREEMAINS (70) SMDCC 348
PPA_DOMAIN_GETMAINS 348
PPA_DOMID (6) SMDCC 347
PPA_DSA_NAME (10) SMDCC 347
PPA_EXTENT_MULTIPLE (34) SMDCC 348
PPA_EXTENT_ROUND (38) SMDCC 348
PPA_EXTENTS (C4) SMDCC 348
PPA_EXTENTS_ADDED (C8) SMDCC 348
PPA_EXTENTS_RELEASED (CC) SMDCC 348
PPA_FLAGS (68) SMDCC 348
PPA_FREE_BYTES (40) SMDCC 348
PPA_FREEHEAD (B8) SMDCC 348
PPA_GETMAINS_NOSTG (8C) SMDCC 348
PPA_HWM_FREE_BYTES (A0) SMDCC 348
PPA_HWM_SIZE (BC) SMDCC 348
PPA_HWM_SUSPENDED (5C) SMDCC 348
PPA_INDEX (6A) SMDCC 348
PPA_LARGEST_FREE_AREA (50) SMDCC 348
PPA_LAST_NOTIFY_FREE_BYTES (48) SMDCC 348
PPA_LENGTH (0) SMDCC 347
PPA_LWM_FREE_BYTES (4C) SMDCC 348
PPA_LWM_SIZE (C0) SMDCC 348
PPA_NEXT 347
PPA_NOTIFY_THRESHOLD (B0) SMDCC 348
PPA_PAGEROUND (2C) SMDCC 347
PPA_PAGESIZE (28) SMDCC 347
PPA_PAGESIZE_SHIFT (D4) SMDCC 348
PPA_PPX_FIRST 347
PPA_PPX_LAST (24) SMDCC 347
PPA_PREFIX (0) SMDCC 347
PPA_PREV (1C) SMDCC 347
PPA_PRIMARY_EXTENT_SIZE (30) SMDCC 348
PPA_REQUESTED_CUSHION_SIZE (D0) SMDCC 348
PPA_REQUESTS_PURGED (64) SMDCC 348
PPA_RESUMED (60) SMDCC 348
PPA_SIZE (B4) SMDCC 348
PPA_SOS (BIT) SMDCC 348
PPA_STORAGE_VIOLATIONS (A4) SMDCC 348
PPA_SUSPENDED (58) SMDCC 348
PPA_SUSPENDS (54) SMDCC 348
PPA_TASK_CUR_PG_STG (80) SMDCC 348
PPA_TASK_FREEMAINS (78) SMDCC 348
PPA_TASK_GETMAINS (74) SMDCC 348
PPA_TASK_HWM_PG_STG (7C) SMDCC 348
PPA_TIME_AT_SOS (98) SMDCC 348
PPA_TIME_WENT_SOS (A8) SMDCC 348
PPA_TIMES_WENT_SOS (94) SMDCC 348
PPTE (0) PGDCC 287
PPTE_ADD_IN_PROGRESS (BIT) PGDCC 288
PPTE_ANY_DATA_LOC (BIT) PGDCC 287
PPTE_ARROW (0) PGDCC 287
PPTE_ARROW_VALUE 1 PGDCC 291
PPTE_ASSEMBLER 4 PGDCC 292
PPTE_ASSEMBLER_CICS (BIT) PGDCC 288
PPTE_AUTOINSTALL 4 PGDCC 291
PPTE_BLOCK_NAME (6) PGDCC 287
PPTE_BLOCK_NAME_VALUE 4 PGDCC 291
PPTE_BUILT_FROM_CATALOG 4 PGDCC 291
PPTE_BUILT_FROM_GROUPLIST 4 PGDCC 291
PPTE_BUILT_FROM_RDO 4 PGDCC 291
PPTE_C370 4 PGDCC 292
PPTE_CATALOG_RECORD (0) PGDCC 287
PPTE_CEDF_STATUS (BIT) PGDCC 287
PPTE_CICS_EXEC_KEY (BIT) PGDCC 287
PPTE_CICS_HOLD (BIT) PGDCC 288
PPTE_COBOL 4 PGDCC 292
PPTE_COBOL2 4 PGDCC 292
PPTE_CS_WORD (30) PGDCC 288
PPTE_DEFINED_THREADSAFE (BIT) PGDCC 287
PPTE_DEFINITIONS (17) PGDCC 287
PPTE_DEFINITIONS_2 (18) PGDCC 287
PPTE_DELETE_IN_PROGRESS (BIT) PGDCC 288
PPTE_DFH (1) PGDCC 287
PPTE_DFH_VALUE 3 PGDCC 291
PPTE_DOMID (4) PGDCC 287
PPTE_DOMID_VALUE 2 PGDCC 291
PPTE_DPLSUBSET (BIT) PGDCC 287
PPTE_DYNAMIC_STATUS (BIT) PGDCC 287
PPTE_HOLD_COUNT (38) PGDCC 288
PPTE_INDICATOR_FLAGS (41) PGDCC 288
PPTE_INDICATORS (3C) PGDCC 288
PPTE_INSTALL_TYPE (16) PGDCC 287
PPTE_INTERNAL_FLAGS (32) PGDCC 288
PPTE_INTERNALS (2C) PGDCC 288
PPTE_JVM (BIT) PGDCC 288
PPTE_JVM_CLASS (0) PGDCC 288
PPTE_JVM_CLASS_DATA (2) PGDCC 288
PPTE_JVM_CLASS_LENGTH (0) PGDCC 288
PPTE_JVM_CLASS_PTR (48) PGDCC 288
PPTE_JVM_DEBUG (BIT) PGDCC 288
PPTE_JVM_LANG 4 PGDCC 292
PPTE_JVM_RUNTIME 4 PGDCC 292
PPTE_LANG_DEDUCED (30) PGDCC 288
PPTE_LANG_DEFINED (15) PGDCC 287
PPTE_LANG_TOKEN (2C) PGDCC 288
PPTE_LE370 4 PGDCC 292
PPTE_LE370_RUNTIME 4 PGDCC 292
PPTE_LENGTH (A) PGDCC 287
PPTE_LOAD_STATUS (40) PGDCC 288
PPTE_LOADABLE 4 PGDCC 291
PPTE_LOADER_TOKEN (34) PGDCC 288
PPTE_LOCK_OWNERS_PTA_PTR (44) PGDCC 288
PPTE_LOCKED 4 PGDCC 291
PPTE_MANUAL 4 PGDCC 291
PPTE_MAPSET 4 PGDCC 291
PPTE_MODULE_TYPE (14) PGDCC 287
PPTE_NON_LE370_RUNTIME 4 PGDCC 292
PPTE_NOT_DEDUCED 4 PGDCC 292
PPTE_NOT_DEFINED 4 PGDCC 292
PPTE_NOT_LOADABLE 4 PGDCC 291
PPTE_NOT_LOADED 4 PGDCC 291

PPTE_PARTITIONSET 4 PGDCC 291
PPTE_PG_CATALOGED_PDB (BIT) PGDCC 288
PPTE_PGWE (BIT) PGDCC 288
PPTE_PLI 4 PGDCC 292
PPTE_PREFIX (0) PGDCC 287
PPTE_PREFIX_VALUE 10 PGDCC 291
PPTE_PROG_ENABLED (BIT) PGDCC 287
PPTE_PROGRAM 4 PGDCC 291
PPTE_PROGRAM_LOCK (31) PGDCC 288
PPTE_PROGRAM_NAME (C) PGDCC 287
PPTE_RELOAD_YES (BIT) PGDCC 287
PPTE_REMOTE (BIT) PGDCC 287
PPTE_REMOTE_PROGID (1C) PGDCC 288
PPTE_REMOTE_SYSID (24) PGDCC 288
PPTE_REMOTE_TRANID (28) PGDCC 288
PPTE_RUNTIME_ENVIRONMENT 288
PPTE_SYSTEM_AUTOINSTALL 4 PGDCC 291
PPTE_THREADSAFE (BIT) PGDCC 287
PPTE_UNLOCKED 4 PGDCC 291
PPTE_USECOUNT (3C) PGDCC 288
PPX (0) SMDCC 349
PPX_ARROW (2) SMDCC 349
PPX_BLOCK_NAME (8) SMDCC 349
PPX_DFH (3) SMDCC 349
PPX_DOMID (6) SMDCC 349
PPX_DSA_NAME (10) SMDCC 349
PPX_EXTENT_END (28) SMDCC 349
PPX_EXTENT_SIZE 349
PPX_EXTENT_START (24) SMDCC 349
PPX_FLAGS (30) SMDCC 349
PPX_FREE_BYTES (40) SMDCC 349
PPX_LENGTH (0) SMDCC 349
PPX_NEXT (18) SMDCC 349
PPX_PAM_BYTES (38) SMDCC 349
PPX_PAM_START (50) SMDCC 349
PPX_PAMP (34) SMDCC 349
PPX_PPAP (3C) SMDCC 349
PPX_PREFIX (0) SMDCC 349
PPX_PREV (1C) SMDCC 349
PPX_PRIMARY (BIT) SMDCC 349
PPX_SAEP (2C) SMDCC 349
PR_READONLY (BIT) BAACT 6, 7
PRCM_GATE (28) PRS 296
PRE_INIT_COMPLETE_FLAG (BIT) MEPS 257
PRE_INITIALISED 4 MEPS 259
PRE_INITIALISED 4 SMDCC 362
PRE_INITIALISED 4 XMANC 437
PRE_INITIALISING 4 SMDCC 362
PRE_INITIALISING 4 XMANC 437
PRE_KEYPOINT (30) RMLI 304
PRE_KEYPOINT (8D8) RMLK 307
PRE_KEYPOINT (98) RMUW 338
PREF_TASK_CICS24 1 SMDCC 356
PREF_TASK_CICS31 1 SMDCC 356
PREF_TASK_USER24 1 SMDCC 356
PREF_TASK_USER31 1 SMDCC 356
PREFIX (0) CPSPS 34
PREFIX (0) PRS 296
PREFIX (0) PTE 298
PREINITIALISED 1 DDCBC 37
PREINITIALISING 1 DDCBC 37
PRELOGGING (9E8) RMLK 308
PRELOGGING (E0) RMLK 310
PRELOGGING_REQUIRED (BIT) RMLK 308, 310
PREPARE_TO_RECEIVE_TYPE 33
PRESUMPTION (14) RMLK 317
PRESUMPTION (68) RMLK 310
PRESUMPTION (970) RMLK 308
PREV (18) RMLS 318
PREV (20) BAACT 9
PREV (20) L2BS 212, 218
PREV (20) L2CH 220, 222
PREV (20) L2SR 243, 249, 250
PREV (20) RMLK 309, 311
PREV (20) RMNS 322
PREV (20) RMUW 330, 335
PREV (278) L2BS 217
PREV (28) RMLK 305
PREV (28) RMLS 318
PREV (28) RMNS 323
PREV (28) RMUW 337
PREV (30) BAACT 9
PREV (30) L2BS 218
PREV (30) L2CH 222
PREV (30) L2SR 249, 250
PREV (30) RMLK 309
PREV (30) RMNS 322
PREV (38) RMLK 305
PREV (38) RMNS 323
PREV (38) RMSL 327, 329
PREV (38) RMUW 337
PREV (48) RMSL 327, 329
PREV (50) RMLK 316
PREV (58) L2BS 213
PREV (58) L2SR 244
PREV (60) RMLK 316
PREV (68) L2BS 213
PREV (68) L2SR 244
PREV (70) L2CH 223
PREV (70) RMUW 338
PREV (78) BAACT 6
PREV (78) RMLK 312
PREV (78) RMUW 332
PREV (8) BAACT 21
PREV (8) L2CH 221
PREV (8) L2SR 251
PREV (8) RMID 303
PREV (8) RMLI 304
PREV (8) RMLK 315
PREV (8) RMNM 321
PREV (8) RMNS 322
PREV (8) RMUW 334, 335
PREV (84) BAACT 13, 19
PREV (88) BAACT 6
PREV (88) RMLK 312
PREV (88) RMUW 332
PREV (8B0) RMLK 306
PREV (928) RMLK 307
PREV (938) RMLK 307
PREV (950) RMUW 339
PREV (960) RMUW 339
PREV (98) L2BS 213
PREV (98) L2SR 244
PREV (A8) L2BS 213
PREV (A8) L2SR 244
PREV (C8) BAACT 18
PREV (D0) RMLK 313
PREV (D0) RMUW 333, 338
PREV (D8) BAACT 18
PREV (E0) RMLK 313
PREV (E0) RMUW 333, 338
PREV (E8) BAACT 11
PREV (F8) BAACT 11
PREVIOUS (34) L2BS 212
PREVIOUS (34) L2SR 243
PRFS_GATE (24) PRS 296
PRI_ALLIGN 4 DSTSK 67
PRIMARY_BLOCK_ID (2C) L2LF 236
PRIMARY_BLOCK_ID (2C) LGSF 201
PRIMARY_BLOCK_ID (C) L2LF 234
PRIMARY_LOG (30) L2CH 220
PRIMARY_LOG_HISTORY_POINT_INFO (24) L2LF 236
PRIMARY_LOG_HISTORY_POINT_INFO (24) LGSF 201
PRIMARY_LOG_HISTORY_POINT_INFO (4) L2LF 234
PRIMARY_STCK_VALUE (24) L2LF 236
PRIMARY_STCK_VALUE (24) LGSF 201
PRIMARY_STCK_VALUE (4) L2LF 234
PRIMARY_STOKEN (10) L2SL 241
PRIORITY (6B) DSTSK 66
PRIORITY_MULTIPLIER (12) DSANC 54
PRIORITY_TIME_FACTOR (90) DSTSK 66
PRM_ACQUIRE_SUSPEND_TOK_FAILED 2 PRS 297
PRM_ACQUIRED_SUSPEND_TOK 2 PRS 297
PRM_INIT_SUCCEEDED 2 PRS 297
PRM_INIT_TASK_ATTACHED 2 PRS 297
PRM_INIT_TASK_STARTED 2 PRS 297
PRM_LOAD_PRCM_FAILED 2 PRS 297
PRM_LOAD_PRFS_FAILED 2 PRS 297
PRM_LOAD_PRPT_FAILED 2 PRS 297
PRM_LOAD_PRRP_FAILED 2 PRS 297
PRM_LOADED_PRCM 2 PRS 297
PRM_LOADED_PRFS 2 PRS 297
PRM_LOADED_PRPT 2 PRS 297
PRM_LOADED_PRRP 2 PRS 297
PRM_OPEN_FOR_BUSINESS 2 PRS 297
PRM_PARTNER_RECOVERED 2 PRS 297

PRM_PARTNER_RECOVERY_FAILED 2 PRS 297
PRM_SSA (0) PRS 296
PRM_SSA_BLOCK_NAMEI 8 PRS 297
PRM_SSA_LENGTH 1 PRS 297
PRM_STATIC_STORAGE_INITIALIZED 2 PRS 297
PRO_ADD 7
PRO_ID 5, 6, 7, 10, 12, 13, 14, 15, 16, 17, 18
PRO_INSTORE 6, 7
PRO_KEY (0) BAACT 7
PRO_LR_KEY (78) BAACT 15
PRO_LR_KEY (8) BAACT 14, 15
PRO_NAME (12) BAACT 14, 15
PRO_NAME (16) BAACT 12, 18
PRO_NAME (1A) BAACT 6, 8
PRO_NAME (2A) BAACT 5, 10
PRO_NAME (3C) BAACT 17
PRO_NAME (44) BAACT 15
PRO_NAME (5C) BAACT 10
PRO_NAME (82) BAACT 15
PRO_NAME (A) BAACT 7, 13, 17
PRO_NAME (B4) BAACT 16
PRO_NAME (E) BAACT 14
PROC_FILE (0) BAACT 14, 15
PROC_FILE (70) BAACT 15
process
  bam process class, BAACT 5
PROCESS (0) BAACT 5
PROCESS_RECORD 6, 7
PROCESS_REF (0) BAACT 7
processtype
  bam processtype class, BAPT 23
PROCESSTYPE (0) BAPT 23
PROFILE_NAME (0) PTE 299
PROFILE_NAME (18) PTE 298
PROFILE_NAME (BC) CPCPS 33
PROFORMA_LINK (908) RMLK 307
PROFORMA_UOW_POINTER (10) RMUW 337
program
  program manager control blocks, PGDCC 286
  statistics utility program anchor block, STUCB 375
  web error program parms, WBEPC 419
PROGRAM (0) BAACT 13
PROGRAM (100) BAACT 11
PROGRAM (E0) BAACT 18
PROGRAM_CHECK_ADDRESS 4
PROGRAM_CHECK_INTERRUPT_DATA (18C) APLI 4
PROGRAM_CHECK_PSW (184) APLI 4
PROGRAM_DEFINITION 8 LDCBS 175
PROGRAM_POOLS_BDY 2 LDCBS 175
PROGRAM24_POOL 4 LDCBS 174
PROGRAM24_POOL_NAME 8 LDCBS 175
PROGRAM24_RO_POOL 4 LDCBS 174
PROGRAM24_RO_POOL_NAME 8 LDCBS 175
PROGRAM31_POOL 4 LDCBS 174
PROGRAM31_POOL_NAME 8 LDCBS 175
PROGRAM31_RO_POOL 4 LDCBS 174
PROGRAM31_RO_POOL_NAME 8 LDCBS 175
programming
  frontend programming interface trace, FEP01 108
  frontend programming interface, FEP21 148
properties
  properties list, FEP12 135
property
  property set info, FEP13 136
PROTOCOL (20) SOA 371
PROTYPE_NAME (18) BAACT 5
PRPT_GATE (20) PRS 296
PRS 296
PRVMOD_PTR (17C) LDCBS 170
PSTORE (10) RMNS 323
PSW (270) APLI 4
PT_BLOCK_NAME_VALUE 4 BAPT 24
PTA (0) PGDCC 289
PTA_ARROW (2) PGDCC 289
PTA_AUTOINSTALL_CALLED (BIT) PGDCC 289
PTA_BLOCK_NAME (8) PGDCC 289
PTA_COMMAREA_RETURNED (BIT) PGDCC 289
PTA_DFH (3) PGDCC 289
PTA_DOMID (6) PGDCC 289
PTA_FLAGS (40) PGDCC 289
PTA_HANDLE_ABEND_CT 289
PTA_INPUTMSG_RETURNED (BIT) PGDCC 289
PTA_JVM_CALLED (BIT) PGDCC 289
PTA_LENGTH (0) PGDCC 289
PTA_LEVEL_COUNTS (38) PGDCC 289
PTA_LOGICAL_LEVEL (38) PGDCC 289
PTA_PLCB_HEAD (18) PGDCC 289
PTA_PREFIX (0) PGDCC 289
PTA_PSEUDO_CONV_COMMAREA (BIT) PGDCC 289
PTA_SYSTEMEXIT_LEVEL (3C) PGDCC 289
PTA_TASK_LLE_HEAD (10) PGDCC 289
PTA_XCTL_ENTRY_POINT (2C) PGDCC 289
PTA_XCTL_INFO (1C) PGDCC 289
PTA_XCTL_LANGUAGE_TOKEN (34) PGDCC 289
PTA_XCTL_LOAD_POINT (28) PGDCC 289
PTA_XCTL_PROG_PPTE (24) PGDCC 289
PTA_XCTL_PROGRAM_LENGTH (30) PGDCC 289
PTA_XCTL_PROGRAM_NAME (1C) PGDCC 289
PTE 297
PTE (0) PTE 298
PTE_BLOCK_NAMEI 8 PTE 299
PTT_DIRECTORY_TOKEN (10) BAPT 23
PTYPE 12
PTYPE_NAME (12) BAACT 6, 7
PTYPE_NAME (2) BAACT 7, 13, 17
PTYPE_NAME (22) BAACT 5, 10
PTYPE_NAME (34) BAACT 17
PTYPE_NAME (3C) BAACT 15
PTYPE_NAME (54) BAACT 10
PTYPE_NAME (6) BAACT 14
PTYPE_NAME (7A) BAACT 15
PTYPE_NAME (A) BAACT 14, 15
PTYPE_NAME (AC) BAACT 16
PTYPE_NAME (E) BAACT 12, 18
PULLED_AND_RECOVERY_SET (BIT) DSTSK 66
PURGE_PENDING 1 DSTSK 67
PURGE_STATUS (45) DSTSK 65
PURGE_TYPE (25) DSTSK 64, 67
PURGEABLE (BIT) DSTSK 66

Q

QAB (0) TSOL 394
QAB_FLAGS (28) TSOL 395
QAB_LOG_BUFFER (3C) TSOL 395
QAB_LOG_BUFFER_HEADER (2C) TSOL 395
QAB_LOG_BUFFER_LENGTH 4 TSOL 396
QAB_MDB_FIRST (20) TSOL 394
QAB_MDB_LAST (24) TSOL 394
QAB_MDBHEAD (20) TSOL 394
QAB_NEXT (0) TSOL 394
QAB_PREFIX (0) TSOL 394
QAB_PREV (4) TSOL 394
QAB_QOB_FIRST (18) TSOL 394
QAB_QOB_LAST (1C) TSOL 394
QAB_QOBHEAD (18) TSOL 394
QAB_SHUNTED (BIT) TSOL 395
QAB_TASK_TOKEN (10) TSOL 394
QAB_TRANSACTION_NUMBER (14) TSOL 394
QAB_UNSHUNTED (BIT) TSOL 395
QAB_UOWID (8) TSOL 394
QBUF_LENGTH 4 L2HS 231
QBUFVERNUM 4 L2HS 231
QLR 397
QLR_COMMITTED_ITEMS (2E) TSQU 398
QLR_CREATION_TIME (18) TSQU 398
QLR_FIRST_OPERATION 398
QLR_FLAGS (32) TSQU 398
QLR_IC_DATA (48) TSQU 398
QLR_IC_DATA_N 398
QLR_LAST_REFERENCED_TIME (20) TSQU 398
QLR_LENGTH (0) TSQU 398
QLR_OLD_CREATION_TIME (40) TSQU 398
QLR_OLD_IC_DATA_N (38) TSQU 398
QLR_PREV_OFFSET (2) TSQU 398
QLR_QUEUE_NAME (8) TSQU 398
QLR_READ_CURSOR (30) TSQU 398
QLR_RECORD_TYPE (4) TSQU 398
QLR_TOTAL_ITEMS (2C) TSQU 398
QLR_TRANSID (28) TSQU 398
QOB (0) TSOL 395
QOB_NEXT (0) TSOL 395
QOB_NQTOKEN (28) TSOL 395
QOB_PREFIX (0) TSOL 395
QOB_PREV (4) TSOL 395

QOB_QABP (20) TSOL 395
QOB_QTOKEN (24) TSOL 395
QOB_QUEUE_NAME (8) TSOL 395
QOB_WAITQ (18) TSOL 395
QPF (0) SMDCC 353
QPF_NEXT (4) SMDCC 353
QPF_SCAP (0) SMDCC 353
QPH (0) SMDCC 353
QPH_ARROW (2) SMDCC 353
QPH_BLOCK_NAME (8) SMDCC 353
QPH_DFH (3) SMDCC 353
QPH_DOMID (6) SMDCC 353
QPH_DONT_FREE_PAGE 353
QPH_FIRST_FREE_CELL (24) SMDCC 353
QPH_FLAGS (2A) SMDCC 353
QPH_LENGTH (0) SMDCC 353
QPH_NAME (10) SMDCC 353
QPH_NEXT (18) SMDCC 353
QPH_NEXT_FREE 353
QPH_NUMBER_FREE_CELLS (28) SMDCC 353
QPH_ON_FREE_CHAIN (BIT) SMDCC 353
QPH_PREFIX (0) SMDCC 353
QPH_PREV (1C) SMDCC 353
QPH_SCAP (2C) SMDCC 353
QR_CPU_PERCENT (168) DSANC 56
QUB (0) TSQU 397
QUB_ITEM_NUMBER (8) TSQU 397
QUB_NEXT (0) TSQU 397
QUB_OLD_ITEMT (C) TSQU 397
QUB_PREV (4) TSQU 397
QUB_TSIP (10) TSQU 397
queue
  domain manager wait queue element, DMCB3 50
  enqueue domain queue element area, NQEA 277
  temporary storage queue class, TSQU 396
  temporary storage wait queue class, TSWQ 402
  work queue element, FEP14 138
QUICK_1_ELEM_NEXT (0) LMCB2 207
QUICK_2_ELEM_NEXT (0) LMCB2 207
QUICK_3_ELEM_NEXT (0) LMCB2 207
quickcell
  lock manager domain quickcell headers, LMCB2 206
QUICKCELL_1 (0) LMCB2 206
QUICKCELL_1_ARROW (2) LMCB2 206
QUICKCELL_1_BLOCK_NAME (8) LMCB2 206
QUICKCELL_1_DFH (3) LMCB2 206
QUICKCELL_1_DOMID (6) LMCB2 206
QUICKCELL_1_ELEMENT (0) LMCB2 207
QUICKCELL_1_LAST_ELEMENT (14) LMCB2 206
QUICKCELL_1_LENGTH (0) LMCB2 206
QUICKCELL_1_NEXT (10) LMCB2 206
QUICKCELL_1_PREFIX (0) LMCB2 206
QUICKCELL_2 (0) LMCB2 206
QUICKCELL_2_ARROW (2) LMCB2 206
QUICKCELL_2_BLOCK_NAME (8) LMCB2 206
QUICKCELL_2_DFH (3) LMCB2 206
QUICKCELL_2_DOMID (6) LMCB2 206
QUICKCELL_2_ELEMENT (0) LMCB2 207
QUICKCELL_2_LENGTH (0) LMCB2 206
QUICKCELL_2_NEXT (10) LMCB2 206
QUICKCELL_2_PREFIX (0) LMCB2 206
QUICKCELL_3 (0) LMCB2 206
QUICKCELL_3_ARROW (2) LMCB2 206
QUICKCELL_3_BLOCK_NAME (8) LMCB2 207
QUICKCELL_3_DFH (3) LMCB2 206
QUICKCELL_3_DOMID (6) LMCB2 206
QUICKCELL_3_ELEMENT (0) LMCB2 207
QUICKCELL_3_LENGTH (0) LMCB2 206
QUICKCELL_3_NEXT (10) LMCB2 207
QUICKCELL_3_PREFIX (0) LMCB2 206
QUICKMAX_1 4 LMCB2 208
QUICKMAX_3 4 LMCB2 208
quiesce
  file control quiesce receive element, FCQRE 104
  file control quiesce send element, FCQSE 105
QUIESCE_IN_PROGRESS (1D) RMSL 327, 329
QUIESCE_IN_PROGRESS (BIT) DSANC 54
QUIESCE_STATS_COLL (7A3) DMCB1 47
QUIESCED 1 DDCBC 37
QUIESCED 4 SMDCC 362
QUIESCED 4 TSA 381
QUIESCED 4 XMANC 437
QUIESCING 241
QUIESCING 4 MEPS 259
QUIESCING 4 SMDCC 362
QUIESCING 4 TSA 381
QUIESCING 4 XMANC 437
QUOTE_FOUND (BIT) PAA 283

R

R0 (0) CAUTR 28
R1 (0) CAUTR 28
R2 (0) CAUTR 28
RABN_ACTION_LIST (18) RRAB 342
RABN_ACTION_LIST_END (1C) RRAB 342
RABN_ATOM_ID (C) RRAB 342
RABN_BACKED_OUT (BIT) RRAB 342
RABN_BITS (15) RRAB 342
RABN_FWD_PTR (8) RRAB 342
RABN_HEADER (0) RRAB 342
RABN_NAME 8 RRAB 343
RCT_ABORT_COUNT (450) D2GLB 89
RCT_ABORT_COUNT (518) D2GLB 91
RCT_ABORT_COUNT (90) D2ENT 82, 84
RCT_ACCOUNT_NONE (BIT) D2ENT 81, 83
RCT_ACCOUNT_NONE (BIT) D2GLB 88, 90
RCT_ACCOUNT_PER_TASK (BIT) D2ENT 81, 83
RCT_ACCOUNT_PER_TASK (BIT) D2GLB 88, 90
RCT_ACCOUNT_PER_TXID (BIT) D2ENT 81, 83
RCT_ACCOUNT_PER_TXID (BIT) D2GLB 88, 90
RCT_ACCOUNT_PER_UOW (BIT) D2ENT 81, 83
RCT_ACCOUNT_PER_UOW (BIT) D2GLB 88, 90
RCT_ACCOUNTREC 81, 83, 88, 90
RCT_ACTIVE_THREAD_CHAIN (46C) D2GLB 89
RCT_ACTIVE_THREAD_CHAIN (534) D2GLB 91
RCT_ACTIVE_THREAD_CHAIN (AC) D2ENT 82, 84
RCT_AUTH_COUNT (448) D2GLB 89
RCT_AUTH_COUNT (510) D2GLB 91
RCT_AUTH_COUNT (88) D2ENT 82, 84
RCT_AUTHID (38) D2ENT 81, 83
RCT_AUTHID (3F8) D2GLB 88
RCT_AUTHID (4C0) D2GLB 90
RCT_AUTHTYPE (40) D2ENT 81, 83
RCT_AUTHTYPE (400) D2GLB 88
RCT_AUTHTYPE (4C8) D2GLB 90
RCT_AUTHTYPE_GROUP (BIT) D2ENT 81, 83
RCT_AUTHTYPE_GROUP (BIT) D2GLB 88, 90
RCT_AUTHTYPE_OPID (BIT) D2ENT 81, 83
RCT_AUTHTYPE_OPID (BIT) D2GLB 88, 90
RCT_AUTHTYPE_SIGNID (BIT) D2ENT 81, 83
RCT_AUTHTYPE_SIGNID (BIT) D2GLB 88, 90
RCT_AUTHTYPE_TERM (BIT) D2ENT 81, 83
RCT_AUTHTYPE_TERM (BIT) D2GLB 88, 90
RCT_AUTHTYPE_TXID (BIT) D2ENT 81, 83
RCT_AUTHTYPE_TXID (BIT) D2GLB 88, 90
RCT_AUTHTYPE_USERID (BIT) D2ENT 81, 83
RCT_AUTHTYPE_USERID (BIT) D2GLB 88, 90
RCT_CALL_COUNT (444) D2GLB 89
RCT_CALL_COUNT (50C) D2GLB 91
RCT_CALL_COUNT (84) D2ENT 82, 84
RCT_COMMIT_COUNT (44C) D2GLB 89
RCT_COMMIT_COUNT (514) D2GLB 91
RCT_COMMIT_COUNT (8C) D2ENT 82, 84
RCT_CSUB_ADDRESS (34) D2ENT 81, 83
RCT_CSUB_ADDRESS (3F4) D2GLB 88
RCT_CSUB_ADDRESS (4BC) D2GLB 90
RCT_CURRENT_ACTIVE_THREADS (420) D2GLB 89
RCT_CURRENT_ACTIVE_THREADS (4E8) D2GLB 90
RCT_CURRENT_ACTIVE_THREADS (60) D2ENT 82, 84
RCT_CURRENT_PROTECTED_THREADS (428) D2GLB 89
RCT_CURRENT_PROTECTED_THREADS (4F0) D2GLB 90
RCT_CURRENT_PROTECTED_THREADS (68) D2ENT 82, 84
RCT_DISABLE_AREA (464) D2GLB 89
RCT_DISABLE_AREA (52C) D2GLB 91
RCT_DISABLE_AREA (A4) D2ENT 82, 84
RCT_DISABLE_ECB (464) D2GLB 89
RCT_DISABLE_ECB (52C) D2GLB 91
RCT_DISABLE_ECB (A4) D2ENT 82, 84
RCT_DISABLE_WAIT_COUNT (465) D2GLB 89
RCT_DISABLE_WAIT_COUNT (52D) D2GLB 91
RCT_DISABLE_WAIT_COUNT (A5) D2ENT 82, 84
RCT_DISABLED (BIT) D2ENT 82, 83
RCT_DISABLED (BIT) D2GLB 88, 90
RCT_DISABLED_ABEND_TRANS (BIT) D2ENT 82, 84

RCT_DISABLED_ABEND_TRANS (BIT) D2GLB 88, 90
RCT_DISABLED_BAD_SQLCODE (BIT) D2ENT 82, 84
RCT_DISABLED_BAD_SQLCODE (BIT) D2GLB 88, 90
RCT_DISABLED_ROUTE_TO_POOL (BIT) D2ENT 82, 83
RCT_DISABLED_ROUTE_TO_POOL (BIT) D2GLB 88, 90
RCT_DISABLING (BIT) D2ENT 82, 83
RCT_DISABLING (BIT) D2GLB 88, 90
RCT_DROLLBACK 81, 83, 88, 90
RCT_DROLLBACK_YES (BIT) D2ENT 81, 83
RCT_DROLLBACK_YES (BIT) D2GLB 88, 90
RCT_DYNAMIC_PLAN_EXIT_ANCHOR (468) D2GLB 89
RCT_DYNAMIC_PLAN_EXIT_ANCHOR (530) D2GLB 91
RCT_DYNAMIC_PLAN_EXIT_ANCHOR (A8) D2ENT 82, 84
RCT_ENABLED_STATUS 82, 83, 88, 90
RCT_EYE (2) D2ENT 81, 83
RCT_EYE (3C2) D2GLB 88
RCT_EYE (48A) D2GLB 89
RCT_FREE_PROT_THREAD_CHAIN (470) D2GLB 89
RCT_FREE_PROT_THREAD_CHAIN (538) D2GLB 91
RCT_FREE_PROT_THREAD_CHAIN (B0) D2ENT 82, 84
RCT_FREE_TCB_CHAIN (474) D2GLB 89
RCT_FREE_TCB_CHAIN (53C) D2GLB 91
RCT_FREE_TCB_CHAIN (B4) D2ENT 83, 84
RCT_LEN (0) D2ENT 81, 83
RCT_LEN (3C0) D2GLB 88
RCT_LEN (488) D2GLB 89
RCT_LOT_CHAIN (478) D2GLB 89
RCT_LOT_CHAIN (540) D2GLB 91
RCT_LOT_CHAIN (B8) D2ENT 83, 84
RCT_MAX_PROTECTED_THREADS (41C) D2GLB 89
RCT_MAX_PROTECTED_THREADS (4E4) D2GLB 90
RCT_MAX_PROTECTED_THREADS (5C) D2ENT 82, 84
RCT_NAME (10) D2ENT 81, 83
RCT_NAME (3D0) D2GLB 88
RCT_NAME (498) D2GLB 89
RCT_PLAN (20) D2ENT 81, 83
RCT_PLAN (3E0) D2GLB 88
RCT_PLAN (4A8) D2GLB 90
RCT_PLANEXIT_NAME (28) D2ENT 81, 83
RCT_PLANEXIT_NAME (3E8) D2GLB 88
RCT_PLANEXIT_NAME (4B0) D2GLB 90
RCT_PREFIX (0) D2ENT 81, 83
RCT_PREFIX (3C0) D2GLB 88
RCT_PREFIX (488) D2GLB 89
RCT_PRIORITY 81, 83, 88, 90
RCT_PRIORITY_EQUAL (BIT) D2ENT 81, 83
RCT_PRIORITY_EQUAL (BIT) D2GLB 88, 90
RCT_PRIORITY_HIGH (BIT) D2ENT 81, 83
RCT_PRIORITY_HIGH (BIT) D2GLB 88, 90
RCT_PRIORITY_LOW (BIT) D2ENT 82, 83
RCT_PRIORITY_LOW (BIT) D2GLB 88, 90
RCT_PROTECTED_THREADS (428) D2GLB 89
RCT_PROTECTED_THREADS (4F0) D2GLB 90
RCT_PROTECTED_THREADS (68) D2ENT 82, 84
RCT_PROTECTED_THREADS_HWM (42C) D2GLB 89
RCT_PROTECTED_THREADS_HWM (4F4) D2GLB 90
RCT_PROTECTED_THREADS_HWM (6C) D2ENT 82, 84
RCT_READYQ 83, 84, 89, 91
RCT_READYQ_COUNT (438) D2GLB 89
RCT_READYQ_COUNT (500) D2GLB 91
RCT_READYQ_COUNT (78) D2ENT 82, 84
RCT_READYQ_HWM (43C) D2GLB 89
RCT_READYQ_HWM (504) D2GLB 91
RCT_READYQ_HWM (7C) D2ENT 82, 84
RCT_READYQ_LOT_CHAIN (480) D2GLB 89
RCT_READYQ_LOT_CHAIN (548) D2GLB 91
RCT_READYQ_LOT_CHAIN (C0) D2ENT 83, 84
RCT_READYQ_SEC_COUNT (484) D2GLB 89
RCT_READYQ_SEC_COUNT (54C) D2GLB 91
RCT_READYQ_SEC_COUNT (C4) D2ENT 83, 84
RCT_SINGLE_PHASE_COUNT (454) D2GLB 89
RCT_SINGLE_PHASE_COUNT (51C) D2GLB 91
RCT_SINGLE_PHASE_COUNT (94) D2ENT 82, 84
RCT_TAMPER_CHECK1 (408) D2GLB 89
RCT_TAMPER_CHECK1 (48) D2ENT 82, 84
RCT_TAMPER_CHECK1 (4D0) D2GLB 90
RCT_TAMPER_CHECK2 (410) D2GLB 89
RCT_TAMPER_CHECK2 (4D8) D2GLB 90
RCT_TAMPER_CHECK2 (50) D2ENT 82, 84
RCT_TASK_COUNT (440) D2GLB 89
RCT_TASK_COUNT (508) D2GLB 91
RCT_TASK_COUNT (80) D2ENT 82, 84
RCT_THREAD_HWM (424) D2GLB 89
RCT_THREAD_HWM (4EC) D2GLB 90
RCT_THREAD_HWM (64) D2ENT 82, 84
RCT_THREAD_LIMIT (418) D2GLB 89
RCT_THREAD_LIMIT (4E0) D2GLB 90
RCT_THREAD_LIMIT (58) D2ENT 82, 84
RCT_THREAD_REUSE_COUNT (458) D2GLB 89
RCT_THREAD_REUSE_COUNT (520) D2GLB 91
RCT_THREAD_REUSE_COUNT (98) D2ENT 82, 84
RCT_THREAD_TERM_COUNT (45C) D2GLB 89
RCT_THREAD_TERM_COUNT (524) D2GLB 91
RCT_THREAD_TERM_COUNT (9C) D2ENT 82, 84
RCT_THREADS (420) D2GLB 89
RCT_THREADS (4E8) D2GLB 90
RCT_THREADS (60) D2ENT 82, 84
RCT_THREADWAIT 82, 83, 88, 90
RCT_THREADWAIT_NO (BIT) D2ENT 82, 83
RCT_THREADWAIT_NO (BIT) D2GLB 88, 90
RCT_THREADWAIT_POOL (BIT) D2ENT 82, 83
RCT_THREADWAIT_POOL (BIT) D2GLB 88, 90
RCT_THREADWAIT_YES (BIT) D2ENT 82, 83
RCT_THREADWAIT_YES (BIT) D2GLB 88, 90
RCT_TIME (18) D2ENT 81, 83
RCT_TIME (3D8) D2GLB 88
RCT_TIME (4A0) D2GLB 89
RCT_TRANSID (30) D2ENT 81, 83
RCT_TRANSID (3F0) D2GLB 88
RCT_TRANSID (4B8) D2GLB 90
RCT_USE_COUNT (430) D2GLB 89
RCT_USE_COUNT (4F8) D2GLB 91
RCT_USE_COUNT (70) D2ENT 82, 84
RCT_USE_COUNT_HWM (434) D2GLB 89
RCT_USE_COUNT_HWM (4FC) D2GLB 91
RCT_USE_COUNT_HWM (74) D2ENT 82, 84
RCT_USERS (430) D2GLB 89
RCT_USERS (4F8) D2GLB 91
RCT_USERS (70) D2ENT 82, 84
RCT_WAIT_OR_OVERFLOW (460) D2GLB 89
RCT_WAIT_OR_OVERFLOW (528) D2GLB 91
RCT_WAIT_OR_OVERFLOW (A0) D2ENT 82, 84
RCT_WAITERS (438) D2GLB 89
RCT_WAITERS (500) D2GLB 91
RCT_WAITERS (78) D2ENT 82, 84
RDAB 299
RDAB_HEAD (0) RDAB 299
RDAB_INIT 8 RDAB 300
RDAB_LAST_RDUB (24) RDAB 299
RDAB_RDAL (10) RDAB 299
RDAB_RDUB (20) RDAB 299
RDAB_RET_CODE (14) RDAB 299
RDAB_SUBPOOL (28) RDAB 299
RDAB_SUSPEND_TOKEN_INIT (18) RDAB 299
RDAB_SUSPEND_TOKEN_RECOVER (1C) RDAB 299
RDAL_ELEMENT (12) RDAB 299
RDAL_FORWARD_PTR (8) RDAB 299
RDAL_HEADER (0) RDAB 299
RDAL_INIT 8 RDAB 300
RDAL_LENGTH (C) RDAB 299
RDAL_TYPE (10) RDAB 299
RDSA 4 SMDCC 363
RDSA_NAME 5 LDCBS 175
RDSA_NAME 8 SMDCC 363
RDUB 300
RDUB_BWD_RDAB_PTR (C) RDUB 300
RDUB_BWD_RRAB_PTR (14) RDUB 300
RDUB_DUMMY_PTR (20) RDUB 300
RDUB_FLAGS (3C) RDUB 301
RDUB_FWD_RDAB_PTR (8) RDUB 300
RDUB_FWD_RRAB_PTR (10) RDUB 300
RDUB_HEADER (0) RDUB 300
RDUB_LOCK_NAME (2B) RDUB 300
RDUB_LOCK_QUIESCE (BIT) RDUB 301
RDUB_LOCK_SHARED (BIT) RDUB 301
RDUB_LOCK_TABLE (38) RDUB 301
RDUB_LOCK_TYPE (BIT) RDUB 301
RDUB_MAX 4 RDUB 301
RDUB_NAME 8 RDUB 301
RDUB_NAMES (2B) RDUB 300
RDUB_NUMBER (1C) RDUB 300
RDUB_RRAB (18) RDUB 300
RDUB_TASKI (24) RDUB 300
RDUB_TRANI (27) RDUB 300
RE (0) CAUTR 28
READ_LIST_ADDR (20) SOA 371

READ_LIST_LENGTH (1C) SOA 371
READ_ONLY (47) RMLS 319
READ_ONLY (A7) RMLK 313
READ_ONLY (A7) RMUW 332
READ_ONLY (FF) RMLK 314
READ_ONLY (FF) RMUW 333
READABLE (BIT) L2BL 208
READCURSOR (0) L2BL 209
REC_TYPE (0) L2LF 233
REC_TYPE_FORK (0) L2LF 234
REC_TYPE_FORK (20) L2LF 235
REC_TYPE_FORK (20) LGSF 200
REC_TYPE_NORMAL (0) L2LF 234, 235
REC_TYPE_NORMAL (20) L2LF 235, 236
REC_TYPE_NORMAL (20) LGSF 200, 201
REC_TYPE_SEC (0) L2LF 234
REC_TYPE_SEC (20) L2LF 236
REC_TYPE_SEC (20) LGSF 201
REC_TYPE_TRIM (0) L2LF 234
REC_TYPE_TRIM (20) L2LF 236
REC_TYPE_TRIM (20) LGSF 201
REC_TYPE_USER (0) L2LF 234
REC_TYPE_USER (20) L2LF 236
REC_TYPE_USER (20) LGSF 201
receive
  file control quiesce receive element, FCQRE 104
  VTAM receive request block, FEP15 139
RECEIVE_TYPE (64) CPCPS 33
RECONSTRUCTED (BIT) RMLK 312
RECONSTRUCTED (BIT) RMUW 331
record
  bam audit record class, BAAR 22
  domain record, DMCB4 51
  log manager record token class, L2RT 239
  log of logs failure record, LGFL 198
RECORD_COUNT (74) L2CH 221
RECORD_TOKEN (10) L2CH 222
records
  transaction manager catalog records, XMCAT 438
RECORDS_IGNORED (BIT) RMLK 315
RECORDS_IGNORED (BIT) RMRO 325
RECORDS_IGNORED (BIT) RMUW 334
RECORDSTACKELEMENT (0) L2CH 221
RECORDTOKEN (0) L2RT 239
RECOVERED (BIT) L2CH 221
RECOVERED (BIT) RMLK 316
RECOVERED (BIT) RMNM 321
RECOVERED (BIT) RMNS 323
recovery
  recovery manager domain management instance, RMDM 301
  recovery manager identity instance, RMID 303
  recovery manager link class data, RMLK 305
  recovery manager link instance, RMLK 309
  recovery manager link set instance, RMLS 318
  recovery manager loggable object identity instance, RMLI 304
  recovery manager logname class data, RMNM 320
  recovery manager logname instance, RMNM 321
  recovery manager logname set instance, RMNS 322
  recovery manager resource owner instance, RMRO 324
  recovery manager system log class data, RMSL 329
  recovery manager system log instance, RMSL 327
  recovery manager unit of work class data, RMUW 337
  recovery manager unit of work instance, RMUW 330
  resource definition recovery definitions, RRAB 341
RECOVERY_FLAGS 220
RECOVERY_INFO 257
RECOVERY_STATUS (9EA) RMLK 308
RECOVERY_STATUS (E2) RMLK 310
REGISTERS_AT_LAST_CICS_CMD (1D4) APLI 4
REGISTERS_AT_PROGRAM_CHECK (194) APLI 4
REL_ACT_ID (12) BAACT 6, 8
REL_ACT_ID (2) BAACT 7, 13, 17
REL_ACT_ID (22) BAACT 5, 10
REL_ACT_ID (34) BAACT 17
REL_ACT_ID (3C) BAACT 15
REL_ACT_ID (54) BAACT 10
REL_ACT_ID (6) BAACT 14
REL_ACT_ID (7A) BAACT 16
REL_ACT_ID (A) BAACT 14, 15
REL_ACT_ID (AC) BAACT 16
REL_ACT_ID (E) BAACT 12, 18
RELATIVE_PRIORITY (19E) DSANC 57
RELATIVE_PRIORITY (1E) DSANC 60
RELEASE_ENQUEUE 1 NQPL 281
remote
  data tables remote sharing anchor block, DTRPS 72
REMOTE_UOW_STATUS (18) RMLK 317
REMOTE_UOW_STATUS (6C) RMLK 310
REMOTE_UOW_STATUS (974) RMLK 308
REMOVE (15) RMUW 335
REPLY_ELEMENT 1 MEMMS 256
REPLY_GATE (A8) DSTSK 66
REPORT_COUNTS (8E8) STUCB 376
REPORT_DATE (8D4) STUCB 376
REPORT_DD (8D6) STUCB 376
REPORT_HOUR (8DC) STUCB 376
REPORT_MIN (8DE) STUCB 376
REPORT_MM (8D4) STUCB 376
REPORT_REQD_FLAGS (83B) STUCB 375
REPORT_SEC (8E0) STUCB 376
REPORT_TIME (8DC) STUCB 376
REPORT_YYYY (8D8) STUCB 376
REQ (BIT) STUCB 376
REQ_FORGET_STATE (BIT) RMLK 315
REQ_FORGET_STATE (BIT) RMRO 325
REQ_FORGET_STATE (BIT) RMUW 334
REQ_REASON (106) BAACT 16
REQ_TYPE 16
request
  bind request save area, FEP04 116
  request parameter area, FEP17 141
  session control request block, FEP18 145
  sh request routing class, SHRTC 344
  VTAM receive request block, FEP15 139
  web request block class, WRB 427
REQUEST_ACTION (0) BAACT 15
REQUEST_FLAGS (104) BAACT 16
REQUEST_REASON (0) BAACT 15
REQUEST_TYPE (60) DSANC 61
requests
  VTAM requests block, FEP16 140
REQUEUE 1 DSTSK 68
RESERVED (30) CCGD 29
RESET 1 L2SR 251
RESET_OCCURRED (BIT) STUCB 377
RESIDENT_POOLS_BDY 2 LDCBS 175
RESIDENT24_POOL 4 LDCBS 174
RESIDENT24_POOL_NAME 8 LDCBS 175
RESIDENT24_RO_POOL 4 LDCBS 174
RESIDENT24_RO_POOL_NAME 8 LDCBS 175
RESIDENT31_POOL 4 LDCBS 174
RESIDENT31_POOL_NAME 8 LDCBS 175
RESIDENT31_RO_POOL 4 LDCBS 174
RESIDENT31_RO_POOL_NAME 8 LDCBS 175
resource
  adapter resource manager, FEP02 113
  recovery manager resource owner instance, RMRO 324
  resource definition anchor block, RDAB 299
  resource definition recovery definitions, RRAB 341
  resource definition update block, RDUB 300
  temporary storage resource lock class, TSRL 401
  transaction manager resource lock element, XMRLC 440
RESOURCE_LOCK_OWNER (4) XMRLC 440
RESOURCE_LOCK_TOKEN (0) XMRLC 440
RESOURCE_LOCK_WAITERS (0) XMRLC 440
RESOURCE_NAME (C) DSTSK 64, 67
RESOURCE_TYPE (1C) DSTSK 64, 67
REST_OF_STCK (4) FCQSE 106
RESTART_STATE (18) RMSL 327, 329
RESTART_STATE_TYPE 327
RESTORED (BIT) L2CH 220
RESUME_REQUIRED (BIT) RMLK 312
RESUME_REQUIRED (BIT) RMUW 331
RESUMED_EARLY 1 DSTSK 68
RESYNC_SCHEDULED (9F1) RMLK 308
RESYNC_SCHEDULED (E9) RMLK 311
RESYNCH_IN_PROGRESS (BIT) RMLK 312
RESYNCH_IN_PROGRESS (BIT) RMUW 331
RESYNCHRONISATION_IN_PROGRESS (46) RMLS 319
RESYNCHRONISATION_IN_PROGRESS (A6) RMLK 313
RESYNCHRONISATION_IN_PROGRESS (A6) RMUW 332
RESYNCHRONISATION_IN_PROGRESS (FE) RMLK 314
RESYNCHRONISATION_IN_PROGRESS (FE) RMUW 333
RET_ENDACTIVITY (BIT) BAACT 12, 18
RETAIN_ENQUEUE 1 NQPL 281
RETENTION_PERIOD 216, 230, 248

RETRY_ADDRESS (294) APLI 4
RETRY_APPEND 251
RETRY_AX_REGISTERS_ADDR (2A4) APLI 4
RETRY_DATA_VECTOR (294) APLI 4
RETRY_ERRCOUNT (1C8) L2BS 216
RETRY_ERRCOUNT (1C8) L2SR 248
RETRY_ERRCOUNT (D8) L2HS 230
RETRY_ERRCOUNT_INC_DONE (1D8) L2BS 216
RETRY_ERRCOUNT_INC_DONE (1D8) L2SR 248
RETRY_ERRCOUNT_INC_DONE (E8) L2HS 230
RETRY_FP_REGISTERS_ADDR (2A0) APLI 4
RETRY_GP_REGISTERS_ADDR (29C) APLI 4
RETRY_PROGRAM_MASK_ADDR (298) APLI 4
RETRY_PSW (258) APLI 4
RETRY_REGISTERS (218) APLI 4
RETRY_REQUEST (BIT) DSTSK 66
RETRY_SUSPEND_START (88) DSTSK 66
RETRY_SUSPEND_START_IN_SECS (88) DSTSK 66
RETURN_CONTROL (68) CPCPS 33
reusable
  logger reusable extended iliffe vector class, RUEI 343
RF (0) CAUTR 28
RF_FORGET_REQUIRED 0 RMRO 326
RF_FORGOTTEN 0 RMRO 326
RF_RESET 0 RMRO 326
RGN_NAME 5 LDCBS 175
RID (12) BAACT 6, 7
RID (2) BAACT 7, 13, 17
RID (22) BAACT 5, 10
RID (34) BAACT 17
RID (3C) BAACT 15
RID (54) BAACT 10
RID (6) BAACT 14
RID (7A) BAACT 15
RID (A) BAACT 14, 15
RID (AC) BAACT 16
RID (E) BAACT 12, 18
RITE (4) DDBSC 35
RLE (0) XMRLC 440
RLE_EYECATCHER (0) XMRLC 440
RLE_FLAGS (10) XMRLC 440
RLE_NEXT (8) XMRLC 440
RLE_RESOURCE (4) XMRLC 440
RLE_RESUMER (BIT) XMRLC 440
RLE_SUSPEND_TOKEN (C) XMRLC 440
RM_EYE_LEN (0) RMDM 301
RM_EYE_LEN (0) RMNM 320
RM_EYE_LEN (0) RMNS 323
RM_EYE_LEN (0) RMUW 337
RM_EYE_LEN (100) RMUW 339
RM_EYE_LEN (40) RMLK 305
RM_EYE_LEN (40) RMUW 337
RM_EYE_LEN (460) RMLK 306
RM_EYE_LEN (520) RMUW 339
RM_EYE_LEN (8) RMLK 305, 309, 311
RM_EYE_LEN (8) RMSL 327, 329
RM_EYE_LEN (8) RMUW 330
RM_EYE_LEN (880) RMLK 306
RM_EYE_LEN (910) RMLK 307
RM_EYE_OFFSET (102) RMUW 339
RM_EYE_OFFSET (2) RMDM 301
RM_EYE_OFFSET (2) RMNM 320
RM_EYE_OFFSET (2) RMNS 323
RM_EYE_OFFSET (2) RMUW 337
RM_EYE_OFFSET (42) RMLK 305
RM_EYE_OFFSET (42) RMUW 337
RM_EYE_OFFSET (462) RMLK 306
RM_EYE_OFFSET (522) RMUW 339
RM_EYE_OFFSET (882) RMLK 306
RM_EYE_OFFSET (912) RMLK 307
RM_EYE_OFFSET (A) RMLK 305, 309, 311
RM_EYE_OFFSET (A) RMSL 327, 329
RM_EYE_OFFSET (A) RMUW 330
RM_EYE_STRING (104) RMUW 339
RM_EYE_STRING (4) RMDM 301
RM_EYE_STRING (4) RMNM 320
RM_EYE_STRING (4) RMNS 323
RM_EYE_STRING (4) RMUW 337
RM_EYE_STRING (44) RMLK 305
RM_EYE_STRING (44) RMUW 337
RM_EYE_STRING (464) RMLK 306
RM_EYE_STRING (524) RMUW 339
RM_EYE_STRING (884) RMLK 306
RM_EYE_STRING (914) RMLK 307
RM_EYE_STRING (C) RMLK 305, 309, 311
RM_EYE_STRING (C) RMSL 327, 329
RM_EYE_STRING (C) RMUW 330
RMC_DATA (65) RMNM 321
RMC_TOKEN (4) RMLK 317
RMC_TOKEN (58) RMLK 310
RMC_TOKEN (960) RMLK 308
RMCD_CLASSID 4 RMDM 302
RMCI_CLIENT_DATA (0) RMLK 316
RMCI_DOMAIN 316
RMCI_GATE (38) RMLK 316
RMCI_PCHAINNODE (18) RMLK 316
RMCI_PERSISTENT_DATA 316
RMCI_PERSISTENT_DATA_PTR (70) RMLK 316
RMCI_REGISTERED (30) RMLK 316
RMCI_RMNS_PTR (6C) RMLK 316
RMCI_SENT_PLIST_PTR (68) RMLK 316
RMCI_TYPE (31) RMLK 316
RMCI_WAITERS 316
RMCLM_MAX_CLASS 4 L2DM 225
RMCLM_MAX_CLASS 4 RMDM 302
RMCLM_OK 4 L2DM 225
RMCR_CHAIN (28) RMSL 327, 329
RMCR_CHAIN (C0) RMUW 338
RMDM 301
RMDM (0) RMDM 301
RMDM_AUTO_OVERRIDE (F0) RMDM 302
RMDM_AUTO_OVERRIDE_TIME (F8) RMDM 302
RMDM_CLASS_MANAGER (1C) RMDM 301
RMDM_CLASSID_SPARE2 4 RMDM 302
RMDM_CLASSID_SPARE3 4 RMDM 302
RMDM_CLASSID_SPARE4 4 RMDM 302
RMDM_CLEAR_LOG_AT_COLD_START (AE) RMDM 301
RMDM_COLD_COPIED (BIT) RMDM 302
RMDM_COLD_COPY_TIME (100) RMDM 302
RMDM_CURR_START_ALL (AD) RMDM 301
RMDM_CURR_START_INIT (AF) RMDM 301
RMDM_CURR_START_TYPE (AC) RMDM 301
RMDM_DIAGNOSTIC_RUN (110) RMDM 302
RMDM_EYE_CATCHER (0) RMDM 301
RMDM_INITIALISED 4 RMDM 303
RMDM_LAST_COLD_TIME (C6) RMDM 302
RMDM_LAST_EMER_TIME (CE) RMDM 302
RMDM_LAST_INIT_TIME (D6) RMDM 302
RMDM_LOCAL_LU_NAME (B0) RMDM 302
RMDM_LOCK_ERROR_CODE 4 RMDM 302
RMDM_LOCK_FREE 4 RMDM 302
RMDM_LOCK_HELD 4 RMDM 302
RMDM_LOCK_STATUS 302
RMDM_LOCK_TOKEN (18) RMDM 301
RMDM_NEXT_START_ALL (C4) RMDM 302
RMDM_NEXT_START_TYPE (C3) RMDM 302
RMDM_NUM_CLASSES 4 RMDM 302
RMDM_OPT_AUTOASIS 8 RMDM 303
RMDM_OPT_AUTOCOLD 8 RMDM 303
RMDM_OPT_AUTODFT 8 RMDM 303
RMDM_OPT_AUTODIAG 8 RMDM 303
RMDM_OPT_AUTOINIT 8 RMDM 303
RMDM_PERSISTENT_DATA (B0) RMDM 301
RMDM_PERSISTENT_OPTIONS 302
RMDM_PNAME 16 RMDM 303
RMDM_POPT_FLAGS (108) RMDM 302
RMDM_POPTIONS_NAME 16 RMDM 303
RMDM_PRE_INITIALISED 4 RMDM 303
RMDM_PRE_INITIALISING 4 RMDM 303
RMDM_PTYPE 8 RMDM 303
RMDM_QUIESCED 4 RMDM 303
RMDM_STATE (C5) RMDM 302
RMDM_SUBPOOL (10) RMDM 301
RMDM_TERMINATED 4 RMDM 303
RMDM_UNLOCK_ERROR_CODE 4 RMDM 302
RMID 303
RMID (0) RMID 303
RMLG_HEADER_LENGTH (0) RMRO 325, 326
RMLG_HEADER_LENGTH (0) RMSL 328
RMLG_HEADER_LENGTH (0) RMUW 335
RMLG_NAME (3) RMRO 325, 326
RMLG_NAME (3) RMSL 328
RMLG_NAME (3) RMUW 335
RMLG_SOURCE (2) RMRO 325, 326
RMLG_SOURCE (2) RMSL 328
RMLG_SOURCE (2) RMUW 335

RMLI 304
RMLI (0) RMLI 304
RMLK 305, 309
RMLK (0) RMLK 309
RMLK_ABENDED 4 RMLK 309, 317
RMLK_CLASS_DATA (0) RMLK 305
RMLK_CLASSID 4 RMDM 302
RMLK_LOGGED_STATE_TYPE (0) RMLK 317
RMLK_LOGGED_TYPE 317
RMLK_MANDATES_LAST 1 RMLK 309, 317
RMLK_ROLLBACK_NOT_SUP 4 RMLK 309, 317
RMLS 318
RMLS (0) RMLS 318
RMLS_ABENDED 4 RMLS 319
RMLS_AWAITING_FORGET 314, 319, 333
RMLS_FAILURE_TIME (112) RMLK 314
RMLS_FAILURE_TIME (112) RMUW 333
RMLS_FAILURE_TIME (5A) RMLS 319
RMLS_FLAGS (111) RMLK 314
RMLS_FLAGS (111) RMUW 333
RMLS_FLAGS (59) RMLS 319
RMLS_LAST_LINK (30) RMLS 318
RMLS_LAST_LINK (E8) RMLK 313
RMLS_LAST_LINK (E8) RMUW 333
RMLS_LINKS (8) RMLS 318
RMLS_LINKS (C0) RMLK 313
RMLS_LINKS (C0) RMUW 332
RMLS_LINKS_INVALID 4 RMLS 319
RMLS_POLLER 313, 318, 333
RMLS_ROLLBACK_NOT_SUPPORTED 4 RMLS 319
RMLS_VOTER (34) RMLS 318
RMLS_VOTER (EC) RMLK 313
RMLS_VOTER (EC) RMUW 333
RMNM 320, 321
RMNM (0) RMNM 321
RMNM_CLASS_DATA (0) RMNM 320
RMNM_CLASS_PNAME 16 RMNM 320, 322
RMNM_CLASS_PNAME 16 RMNS 324
RMNM_CLASSID 4 RMDM 302
RMNM_EYE_CATCHER (0) RMNM 320
RMNM_FLAT_TYPE (0) RMNM 322
RMNM_INSTANCE (8) RMNS 323
RMNM_LOCAL_APPLID (52) RMNM 320
RMNM_LOCAL_LOGNAME (10) RMNM 320
RMNM_PERSISTENT_DATA (10) RMNM 320
RMNM_PSTORE (5A) RMNM 320
RMNM_RMC_DATA_TYPE 322
RMNS 322
RMNS (0) RMNS 322
RMNS_CLASSID 4 RMDM 302
RMNS_INSTANCE (0) RMNS 323
RMNS_RECORD_NAME_TYPE (0) RMNS 323
RMRO 324
RMRO (0) RMRO 324
RMRO_BFAIL_LOG_HDR (0) RMRO 325
RMRO_BFAIL_MEMBER_LOG_HDR (0) RMRO 325
RMRO_BFAILLH_DISCRIMINANT (0) RMRO 325
RMRO_BFAILLH_TYPE (7) RMRO 325
RMRO_BFAILMEMLH_DISCRIMINANT (0) RMRO 325
RMRO_BFAILMEMLH_LOCAL_ACCESS_ID (12) RMRO 325
RMRO_BFAILMEMLH_RESOURCE_ID (8) RMRO 325
RMRO_BFAILMEMLH_TYPE (7) RMRO 325
RMRO_CD_LOG_HDR (0) RMRO 325
RMRO_CDLH_BACKWARD_DATA (BIT) RMRO 325
RMRO_CDLH_DISCRIMINANT (0) RMRO 325
RMRO_CDLH_FLAGS (8) RMRO 325
RMRO_CDLH_FORGET_REQUESTED (BIT) RMRO 325
RMRO_CDLH_FORWARD_DATA (BIT) RMRO 325
RMRO_CDLH_RESOURCE_ID (B) RMRO 325
RMRO_CDLH_RESOURCE_ID_LENGTH (9) RMRO 325
RMRO_CDLH_RESOURCE_ID_X (BIT) RMRO 325
RMRO_CDLH_TYPE (7) RMRO 325
RMRO_CLASSID 4 RMDM 302
RMRO_FO_DISCRIMINANT (0) RMRO 326
RMRO_FO_TYPE (7) RMRO 326
RMRO_FORCE_TOKEN (0) RMRO 325
RMRO_FORGOTTEN_LOG_HDR (0) RMRO 326
RMRO_LOG_RECORD_TYPE (0) RMRO 325
RMRO_REQ_FORGET_LOG_HDR (0) RMRO 325
RMRO_RF_DISCRIMINANT (0) RMRO 325
RMRO_RF_LOCAL_ACCESS_ID (A) RMRO 326
RMRO_RF_LOCAL_ACCESS_ID_LEN (8) RMRO 325
RMRO_RF_TYPE (7) RMRO 325
RMRO_SPARE_NAME 326
RMRO_SYSTEM_LOG_ID_NAME 4 RMRO 326
RMRO_TYPE_BFAIL_BEGIN 1 RMRO 326
RMRO_TYPE_BFAIL_END 1 RMRO 326
RMRO_TYPE_BFAIL_MEMBER 1 RMRO 326
RMRO_TYPE_CLIENT_DATA 1 RMRO 326
RMRO_TYPE_FORGOTTEN 1 RMRO 326
RMRO_TYPE_REQ_FORGET 1 RMRO 326
RMSL 327, 329
RMSL (0) RMSL 327
RMSL_BUFFER_FULL 4 RMSL 328, 330
RMSL_CHAIN (0) RMSL 328
RMSL_CLASS_DATA (0) RMSL 329
RMSL_CLASSID 4 RMDM 302
RMSL_EYE_CATCHER (8) RMSL 327, 329
RMSL_INVALID_DATA_LENGTH 4 RMSL 328, 330
RMSL_LH_DATA (1C) RMSL 328
RMSL_LH_DISCRIMINANT (0) RMSL 328
RMSL_LH_END_OF_COLD_RECOVERY (BIT) RMSL 328
RMSL_LH_END_OF_KEYPOINT (BIT) RMSL 328
RMSL_LH_FLAGS (7) RMSL 328
RMSL_LH_KEYPOINT (BIT) RMSL 328
RMSL_LH_START_OF_COLD_RECOVERY (BIT) RMSL 328
RMSL_LH_START_OF_KEYPOINT (BIT) RMSL 328
RMSL_LH_TASKID (18) RMSL 328
RMSL_LH_TERMID (8) RMSL 328
RMSL_LH_TERMINAL_LUNAME (C) RMSL 328
RMSL_LH_TRANID (14) RMSL 328
RMSL_LOG_HEADER (0) RMSL 328
RMSL_NULL_CHAIN 4 RMSL 328, 330
RMST_CLASSID 4 RMDM 302
RMUW 330, 337
RMUW (0) RMUW 330
RMUW_BUFFER_FULL 4 RMUW 336, 341
RMUW_CLASS_DATA (0) RMUW 337
RMUW_CLASSID 4 RMDM 302
RMUW_CONTEXT (0) RMUW 336
RMUW_CS_COUNT (0) RMUW 336
RMUW_CS_STATES (1) RMUW 336
RMUW_INVALID_DATA_LENGTH 4 RMUW 336, 341
RMUW_LC_FIRST_UOW_FOR_TXN (BIT) RMUW 336
RMUW_LC_FLAGS (42) RMUW 336
RMUW_LC_REMOTE_UOW_ID (1F) RMUW 336
RMUW_LC_TIME (3A) RMUW 336
RMUW_LC_UOW_CONTEXT (0) RMUW 336
RMUW_LH_CHOICE_FORWARD (BIT) RMUW 335
RMUW_LH_CLIENT_STATE_PRESENT (BIT) RMUW 335
RMUW_LH_CONTEXT_PRESENT (BIT) RMUW 335
RMUW_LH_DATA (11) RMUW 335
RMUW_LH_DISCRIMINANT (0) RMUW 335
RMUW_LH_FLAGS (10) RMUW 335
RMUW_LH_HEURISM (BIT) RMUW 335
RMUW_LH_LOCAL_UOW_ID (7) RMUW 335
RMUW_LH_UOW_STATUS (F) RMUW 335
RMUW_LOG_CLIENT_STATE (0) RMUW 336
RMUW_LOG_CONTEXT (0) RMUW 336
RMUW_LOG_HEADER 335
RMUW_LOG_STATUS (0) RMUW 335
RMUW_LS_HEURISTIC_CAUSE (8) RMUW 336
RMUW_LS_TIME (0) RMUW 335
RMUX_CLIENT_STATES (198) RMLK 314
RMUX_CLIENT_STATES (198) RMUW 333
RMUX_FLAGS 314, 333
RMUX_LOCAL_UOW_ID (128) RMLK 314
RMUX_LOCAL_UOW_ID (128) RMUW 333
RMUX_REMOTE_ID_LENGTH (130) RMLK 314
RMUX_REMOTE_ID_LENGTH (130) RMUW 333
RMUX_REMOTE_ID_LU_NAME_LENGTH (131) RMLK 314
RMUX_REMOTE_ID_LU_NAME_LENGTH (131) RMUW 333
RMUX_REMOTE_UOW_ID (130) RMLK 314
RMUX_REMOTE_UOW_ID (130) RMUW 333
RMUX_WORK_TOKEN_ARRAY (14C) RMLK 314
RMUX_WORK_TOKEN_ARRAY (14C) RMUW 333
RMVP_CLASSID 4 RMDM 302
RO_ARRAY (1B0) RMLK 314
RO_ARRAY (1B0) RMUW 333
RO_CLIENT_FLAGS (1D1) RMLK 315
RO_CLIENT_FLAGS (1D1) RMUW 334
RO_CLIENT_FLAGS (21) RMRO 325
ROOT (C) DDBSC 35
ROOT_ACT_REF (20) BAACT 5
routine

routine *(continued)*
  data tables SVC routine anchor blocks, DTSPS 72
routing
  sh request routing class, SHRTC 344
RPL_ARRAY_A (20) CCGD 29
RR_CANCEL_CMD 1 BAACT 20
RR_CANCEL_COMPL 1 BAACT 20
RR_CANCEL_FORCE 1 BAACT 20
RR_DELETE_CMD 1 BAACT 20
RR_DELETE_COMPL 1 BAACT 20
RR_DELETE_RESET 1 BAACT 20
RR_DELETE_TREE 1 BAACT 20
RR_FIRE_COMPL 1 BAACT 20
RR_FIRE_INPUT 1 BAACT 20
RR_FIRE_TIMER 1 BAACT 20
RR_REATTACH_ACQ 1 BAACT 20
RR_UNKNOWN 1 BAACT 20
RRAB 341
RRAB_BITS (28) RRAB 342
RRAB_CURRENT_ACTION_LIST (8) RRAB 341
RRAB_CURRENT_ACTION_LIST_END (C) RRAB 341
RRAB_CURRENT_RABN (14) RRAB 341
RRAB_DELAYED_ACTION_LIST (18) RRAB 342
RRAB_DELAYED_ACTION_LIST_END (1C) RRAB 342
RRAB_FORGET (BIT) RRAB 342
RRAB_HDR (0) RRAB 341
RRAB_LAST_RDUB (24) RRAB 342
RRAB_NAME 8 RRAB 343
RRAB_NAMED_LIST (10) RRAB 341
RRAB_OPEN (BIT) RRAB 342
RRAB_RDUB (20) RRAB 342
RRAB_TOR (BIT) RRAB 342
RRT (BIT) STUCB 376
RS_COLD 4 RMSL 328, 330
RS_COMPLETE 4 RMSL 328, 330
RS_DELIVERY_IN_PROGRESS 4 RMSL 328, 330
RS_DISJOINT 4 RMSL 328, 330
RS_KEYPOINT_DELIVERY 4 RMSL 328, 330
RS_KEYPOINT_IN_PROGRESS 4 RMSL 328, 330
RS_PRE_KEYPOINT 4 RMSL 328, 330
RS_RESET 4 RMSL 328, 330
RSA (0) PGHM 295
RSA_NEXT (44) PGHM 295
RSA_REGS (0) PGHM 295
RSA_USER_COUNT (40) PGHM 295
RTYPE (0) BAACT 7, 13, 17
RTYPE (10) BAACT 6, 7
RTYPE (20) BAACT 5, 10
RTYPE (32) BAACT 17
RTYPE (3A) BAACT 15
RTYPE (4) BAACT 14
RTYPE (52) BAACT 10
RTYPE (78) BAACT 15
RTYPE (8) BAACT 14, 15
RTYPE (AA) BAACT 16
RTYPE (C) BAACT 12, 18
RUEI 343
RUEI (0) RUEI 343
RUEI_BROWSE_END 4 RUEI 344
RUEI_CONTINUATION (10) RUEI 343
RUEI_CONTINUATION_FLAG 343
RUEI_ELEM_ADDR (8) RUEI 343
RUEI_ELEM_ADDR_FLAG 343
RUEI_ELEM_LENGTH (C) RUEI 343
RUEI_ELEM_LENGTH_SUM 343
RUEI_ELEM_LENGTH_SUM_SUM (4) RUEI 343
RUEI_ELEMS (8) RUEI 343
RUNNING_ABTERM_ALLOWED 1 DSTSK 68
RUNNING_ABTERM_NOT_ALLOWED 1 DSTSK 68
RUNNING_TASK (2C) DSANC 59

S

SAE (0) SMDCC 349
SAE_ACCESS 349
SAE_DSA_NAME (7) SMDCC 349
SAE_EXTENT_END (4) SMDCC 349
SAE_PPXP (0) SMDCC 349
SAFPB (0) STAFB 373
SAFPB_ARROW (2) STAFB 373
SAFPB_BLOCK_ID (8) STAFB 373
SAFPB_CREATION_STCK 373
SAFPB_DFH (3) STAFB 373
SAFPB_DOMAIN (6) STAFB 373
SAFPB_FUNCTION (10) STAFB 373
SAFPB_GTF_TRACE_FLAG 373
SAFPB_GTF_TRACE_OFF 0 STAFB 373
SAFPB_GTF_TRACE_ON 0 STAFB 373
SAFPB_INVALID_FUNCTION 1 STAFB 373
SAFPB_INVALID_RECORD_LENGTH 1 STAFB 373
SAFPB_LENGTH (0) STAFB 373
SAFPB_NO_AUTHORISATION 1 STAFB 373
SAFPB_NO_FESTAE 1 STAFB 373
SAFPB_NO_STORAGE_253 1 STAFB 373
SAFPB_NO_STORAGE_SMF 1 STAFB 373
SAFPB_NOT_CICS_RECORD 1 STAFB 373
SAFPB_OK 1 STAFB 373
SAFPB_PREFIX (0) STAFB 373
SAFPB_PTR (54) STCB1 374
SAFPB_RESPONSE (12) STAFB 373
SAFPB_RTNREG0 (20) STAFB 373
SAFPB_RTNREG1 (24) STAFB 373
SAFPB_RTNREG15 (28) STAFB 373
SAFPB_SMF_ERROR 1 STAFB 373
SAFPB_SMF_RC 373
SAFPB_SMF_RECORD 373
SAFPB_SMFEWTM 2 STAFB 373
SAT (0) SMDCC 349
SAT_ABOVE (220) SMDCC 349
SAT_ABOVE_SHIFT (1C) SMDCC 349
SAT_ABOVEP (18) SMDCC 349
SAT_ARROW (2) SMDCC 349
SAT_BELOW (20) SMDCC 349
SAT_BELOW_SHIFT (14) SMDCC 349
SAT_BELOWP (10) SMDCC 349
SAT_BLOCK_NAME (8) SMDCC 349
SAT_DFH (3) SMDCC 349
SAT_DOMID (6) SMDCC 349
SAT_LENGTH (0) SMDCC 349
SAT_PREFIX (0) SMDCC 349
SATBLOCK_NAME 8 SMDCC 356
SATBLOCK_SIZE 4 SMDCC 363
save
  bind request save area, FEP04 116
  macro save area, PGA 285
SAVED_NEXT_TCP_DISPATCH_TIME (160) DSANC 56
SB_EYE_CATCHER (0) DSANC 60
SBB (0) TSRL 400
SBB_FIRST (BIT) TSRL 400
SBB_FLAGS (2C) TSRL 400
SBB_NAME (18) TSRL 400
SBB_NEXT (0) TSRL 400
SBB_PCAP (28) TSRL 400
SBB_PREFIX (0) TSRL 400
SBB_PREV (4) TSRL 400
SBB_TRANID (8) TSRL 400
SBB_TRANNUM (C) TSRL 400
SBB_TRANTOKEN (10) TSRL 400
SCA (0) SMDCC 351
SCA_ACCESS (11) SMDCC 351
SCA_ANY (BIT) SMDCC 351
SCA_BDYROUND (88) SMDCC 351
SCA_BOUNDARY (8C) SMDCC 351
SCA_CLEAR_STG (BIT) SMDCC 351
SCA_DSA_INDEX (12) SMDCC 351
SCA_ELEMCHAIN (90) SMDCC 351
SCA_ELEMENT_STORAGE (9C) SMDCC 351
SCA_ELEMHEAD (50) SMDCC 351
SCA_ELEMTYPE (91) SMDCC 351
SCA_FIRST_FREE_QPH (24) SMDCC 351
SCA_FIRST_QPH (1C) SMDCC 351
SCA_FIXEDLEN (18) SMDCC 351
SCA_FLAGS 351
SCA_FREE_NAME 8 SMDCC 356
SCA_FREEHEAD (60) SMDCC 351
SCA_FREEMAINS (38) SMDCC 351
SCA_FREEZE_STG (BIT) SMDCC 351
SCA_GETMAINS (30) SMDCC 351
SCA_HEAD_NAME 8 SMDCC 356
SCA_HWM_PAGE_STORG (A4) SMDCC 351
SCA_IFA_FIRST (78) SMDCC 351
SCA_IFA_LAST (7C) SMDCC 351
SCA_IFAHEAD (78) SMDCC 351
SCA_INITFREE_LEN1 (80) SMDCC 351
SCA_INITFREE_LEN2 351

SCA_INLINE (BIT) SMDCC 351
SCA_LAST_QPH (20) SMDCC 351
SCA_LOCK_TOKEN (34) SMDCC 351
SCA_MAX_FREE_CELLS_LESS1 351
SCA_MIN_FREE_CELLS (2E) SMDCC 351
SCA_NAME (0) SMDCC 351
SCA_NEXT (8) SMDCC 351
SCA_NUM (70) SMDCC 351
SCA_NUMELEMS_LAST_RESET (A0) SMDCC 351
SCA_OWNER (84) SMDCC 351
SCA_PAGE_STORAGE (98) SMDCC 351
SCA_PPAP (74) SMDCC 351
SCA_PREFIX (0) SMDCC 351
SCA_PREV (C) SMDCC 351
SCA_QUICKCELL (BIT) SMDCC 351
SCA_RESET_STATS (BIT) SMDCC 351
SCA_SELF_TUNING (BIT) SMDCC 351
SCA_SMXP (A8) SMDCC 351
SCA_SPID (8E) SMDCC 351
SCA_STORAGE_CHECK (BIT) SMDCC 351
SCA_SUBSPACE_TOKEN (AC) SMDCC 351
SCA_TUNING_AVERAGE (48) SMDCC 351
SCA_TUNING_INTERVALS (44) SMDCC 351
SCA_USAGE (8F) SMDCC 351
SCABLOCK_NAME 8 SMDCC 356
SCABLOCK_SIZE 4 SMDCC 363
SCAN_DELAY_INTERVAL (18) DSANC 54
SCAN_DELAY_INTERVAL_SIT (68) DSANC 54
SCB (0) SMDCC 352
SCB_ARROW (2) SMDCC 352
SCB_BLOCK_NAME (8) SMDCC 352
SCB_DFH (3) SMDCC 352
SCB_DOMID (6) SMDCC 352
SCB_LENGTH (0) SMDCC 352
SCB_NEXT (10) SMDCC 352
SCB_PREFIX (0) SMDCC 352
SCE (0) SMDCC 353
SCE_ADDR (8) SMDCC 353
SCE_LEN (C) SMDCC 353
SCE_NEXT (0) SMDCC 353
SCE_PPXP 353
SCE_PREFIX (0) SMDCC 353
SCE_PREV (4) SMDCC 353
SCF (0) SMDCC 354
SCF_ADDR (8) SMDCC 354
SCF_LEN (C) SMDCC 354
SCF_NEXT (0) SMDCC 354
SCF_NULL 4 SMDCC 356
SCF_PPXP 354
SCF_PREFIX (0) SMDCC 354
SCF_PREV (4) SMDCC 354
SCQ (0) SMDCC 353
SCQ_NEXT (0) SMDCC 353
SCQBLOCK_NAME 8 SMDCC 356
SCQBLOCK_SIZE 4 SMDCC 363
SD_EYE_CATCHER (0) DSANC 60
SD_EYE_CATCHER (180) DSANC 57
SDSA 4 SMDCC 363
SDSA_NAME 5 LDCBS 175
SDSA_NAME 8 SMDCC 363
SEC_BROWSE (BIT) L2CH 220
SECOND_BLOCK 213, 244
SECONDARY_BLOCK_ID (1C) L2LF 235
SECONDARY_BLOCK_ID (3C) L2LF 236
SECONDARY_BLOCK_ID (3C) LGSF 201
SECONDARY_CHAIN_HEADER (0) L2LF 234
SECONDARY_INITIALISATION 8 LDCBS 175
SECONDARY_LOG_HISTORY_POINT_INFO (14) L2LF 234
SECONDARY_LOG_HISTORY_POINT_INFO (34) L2LF 236
SECONDARY_LOG_HISTORY_POINT_INFO (34) LGSF 201
SECONDARY_RM_START (14) L2LF 234
SECONDARY_RM_START (34) L2LF 236
SECONDARY_RM_START (34) LGSF 201
SECONDARY_STCK_VALUE (14) L2LF 234
SECONDARY_STCK_VALUE (34) L2LF 236
SECONDARY_STCK_VALUE (34) LGSF 201
SECONDARY_STOKEN (14) L2SL 241
security
  data tables security anchor block, DTXPS 74
  security domain anchor block, XSANC 448
  security domain transaction data, XSXD 455
  security domain transaction token, XSXT 456
security *(continued)*
  security supervisor storage, XSSS 451
SEG_ACQUIRED_FROM_SM (BIT) LIFO 203
SEG_ANYWHERE 4 LIFO 204
SEG_BELOW 4 LIFO 204
SEG_CHAIN (C) LIFO 203
SEG_CURRENT_STACK (18) LIFO 203
SEG_DATA 203
SEG_DISPOSABLE (BIT) LIFO 203
SEG_END_OF_SEGMENT (14) LIFO 203
SEG_FLAGS (1C) LIFO 203
SEG_NAME (0) LIFO 203
SEG_NEXT_FREE (8) LIFO 203
SEG_START_OF_SEGMENT (10) LIFO 203
segment
  stack segment table header, LIFO 203
SEGMENT_ADDRESS_LIMIT 4 LIFO 204
SEGMENT_DATA_EXTLEN_24 4 LIFO 204
SEGMENT_DATA_EXTLEN_31 4 LIFO 204
SEGMENT_DATA_LENGTH_24 4 LIFO 204
SEGMENT_DATA_LENGTH_31 4 LIFO 204
SEGMENT_ENTRY (0) LIFO 203
SELECT_AUTOINST (BIT) STUCB 377
SELECT_CONNECT (BIT) STUCB 377
SELECT_DB2 (BIT) STUCB 377
SELECT_DBCTL (BIT) STUCB 377
SELECT_DCE (BIT) STUCB 377
SELECT_DISPATCH (BIT) STUCB 377
SELECT_ENQUEUE 377
SELECT_FEPI (BIT) STUCB 377
SELECT_FILE 377
SELECT_IGNORE_F (BIT) STUCB 377
SELECT_JOURNAL (BIT) STUCB 377
SELECT_LOGSTREAM (BIT) STUCB 377
SELECT_LSRPOOL (BIT) STUCB 377
SELECT_MONITOR (BIT) STUCB 377
SELECT_PARMS (18) SOA 371
SELECT_PROGAUTO (BIT) STUCB 377
SELECT_PROGRAM (BIT) STUCB 377
SELECT_RECOVERY (BIT) STUCB 377
SELECT_STATS (BIT) STUCB 377
SELECT_STORAGE (BIT) STUCB 377
SELECT_SYSDUMP (BIT) STUCB 377
SELECT_TABLEMGR (BIT) STUCB 377
SELECT_TCLASS (BIT) STUCB 377
SELECT_TCPIPSERVICE (BIT) STUCB 377
SELECT_TDQUEUE (BIT) STUCB 377
SELECT_TERMINAL (BIT) STUCB 377
SELECT_TRANDUMP (BIT) STUCB 377
SELECT_TRANSACT (BIT) STUCB 377
SELECT_TSQUEUE (BIT) STUCB 377
SELECT_TYPE_FLAG1 (92F) STUCB 377
SELECT_TYPE_FLAG2 (930) STUCB 377
SELECT_TYPE_FLAG3 (931) STUCB 377
SELECT_TYPE_FLAG4 (932) STUCB 377
SELECT_TYPE_FLAGS 377
SELECT_USER (BIT) STUCB 377
SELECT_VTAM (BIT) STUCB 377
SELECTED_DATE_PERIOD (912) STUCB 377
SELECTED_PERIOD (90C) STUCB 376
SELECTED_TIME_PERIOD (90C) STUCB 377
send
  file control quiesce send element, FCQSE 105
SEND_TYPE (6C) CPCPS 33
SEQ_RETRY_NUMBER (A58) CCGD 30
SEQ_WRITE_NUMBER 30
SERIAL_RECOVERY (BIT) RMLK 312
SERIAL_RECOVERY (BIT) RMUW 331
services
  builder services action blocks, ZCQ 456
  dce services domain global statistics, DEGPC 38
session
  session control request block, FEP18 145
set
  property set info, FEP13 136
  recovery manager link set instance, RMLS 318
  recovery manager logname set instance, RMNS 322
SET (0) WRB 430
SET_CHAIN_TOKEN (28) RMLI 304
SET_CHAIN_TOKEN (8D0) RMLK 307
SET_CHAIN_TOKEN (90) RMUW 338
SET_NO 1 WRB 430
SET_YES 1 WRB 430

SETSOCKOPT_PARMS (18) SOA 372
SETSTGL 4 TSRL 400
sh
  sh request routing class, SHRTC 344
SHA 399
SHA_ARROW (2) TSRL 399
SHA_BLOCK_NAME (8) TSRL 399
SHA_COMPID (6) TSRL 399
SHA_DFH (3) TSRL 399
SHA_LENGTH (0) TSRL 399
SHA_PBB_FIRST (28) TSRL 399
SHA_PBB_LAST (2C) TSRL 399
SHA_PBBHEAD (28) TSRL 399
SHA_PCA_FIRST (18) TSRL 399
SHA_PCA_LAST (1C) TSRL 399
SHA_PCAHEAD (18) TSRL 399
SHA_POOLS_CONNECTED (34) TSRL 399
SHA_POOLS_DEFINED (30) TSRL 399
SHA_PREFIX (0) TSRL 399
SHA_READ_REQUESTS (38) TSRL 399
SHA_SBB_FIRST (20) TSRL 399
SHA_SBB_LAST (24) TSRL 399
SHA_SBBHEAD (20) TSRL 399
SHA_STATISTICS (30) TSRL 399
SHA_STE_FIRST (10) TSRL 399
SHA_STE_LAST (14) TSRL 399
SHA_STEHEAD (10) TSRL 399
SHA_SYSID_TABLE (10) TSRL 399
SHA_WRITE_REQUESTS (3C) TSRL 399
shared
  temporary storage shared class, TSRL 399
sharing
  data tables remote sharing anchor block, DTRPS 72
SHR (0) SMMCC 365
SHR_CLASS (0) SMMCC 365
SHR_DATA (4) SMMCC 365
SHR_INITIMG (1) SMMCC 365
SHR_LENGTH (2) SMMCC 365
SHR_SAA (0) SMMCC 365
SHRTC 344
SHUNTED (BIT) RMLK 312
SHUNTED (BIT) RMUW 331
SHUTDOWN_DISPATCHER 54
SHUTDOWN_TCB (BIT) DSANC 59
SIGPROCMASK_PARMS (18) SOA 372
simulation
  terminal simulation facility, FEP19 146
SINGLE_UPDATER (9E9) RMLK 308
SINGLE_UPDATER (E1) RMLK 310
SIT_LOADED (BIT) PAA 283
SIT_NAME 6 PAA 284
SIT_PTR (2C) PAA 283
SIT_SUFFIX 283
SITNAME (14) PAA 283
SIXTEEN_MEG 4 LDCBS 176
SIZE (4) BAACT 9
SIZE (5C) BAACT 6
SIZE (AC) BAACT 18
SIZE (CC) BAACT 11
SL_PRIMARY 8 L2SL 241
SL_SECONDARY 8 L2SL 241
SL_UH_END (10) L2LF 236
SL_UH_END (10) LGSF 202
SL_UH_TD_LENGTH (0) L2LF 236
SL_UH_TD_LENGTH (0) LGSF 202
SL_UH_TD_TASKNO (4) L2LF 236
SL_UH_TD_TASKNO (4) LGSF 202
SL_UH_TD_TERMID (C) L2LF 236
SL_UH_TD_TERMID (C) LGSF 202
SL_UH_TD_TRANID (8) L2LF 236
SL_UH_TD_TRANID (8) LGSF 202
SL_UH_TRAN_DATA 202
SL_UH_TRAN_DATA (0) L2LF 236
SL_USER_HEADER (0) LGSF 202
SLBH 199
SLBH (0) L2LF 232
SLBH_BLOCK_TYPE_ARROW 1 L2LF 237
SLBH_BLOCK_TYPE_ARROW 1 LGSF 203
SLBH_BLOCK_TYPE_DFH 3 L2LF 237
SLBH_BLOCK_TYPE_DFH 3 LGSF 203
SLBH_BLOCK_VERSION_NO 2 L2LF 237
SLBH_BLOCK_VERSION_NO 2 LGSF 203
SLBH_DATA (34) L2LF 233
SLBH_DATA (34) LGSF 199
SLBH_LAST_USED_INDEX (30) L2LF 233
SLBH_LAST_USED_INDEX (30) LGSF 199
SLBH_LOG_TYPE_GENERAL 1 L2LF 237
SLBH_LOG_TYPE_GENERAL 1 LGSF 203
SLBH_LOG_TYPE_SYSTEM 1 L2LF 237
SLBH_LOG_TYPE_SYSTEM 1 LGSF 203
SLBH_PREV_BLOCK_ID (28) L2LF 233
SLBH_PREV_BLOCK_ID (28) LGSF 199
SLEEP 1 DSTSK 68
SLF_BAD_BLOCK_SIZE 1 L2SL 241
SLF_DATA_NOT_FOUND 1 L2SL 241
SLF_DISASTER 1 L2SL 241
SLF_LOST_ACCESS 1 L2SL 241
SLF_LOST_DATA 1 L2SL 241
SLF_NONE 1 L2SL 241
SLF_NOT_ACTIVE 1 L2SL 241
SLH_FORK (20) L2LF 235
SLH_FORK (20) LGSF 200
SLH_MASTER (10) L2LF 235
SLH_MASTER (10) LGSF 200
SLH_NON_MOVED 201, 236
SLH_NORMAL (20) L2LF 235
SLH_NORMAL (20) LGSF 200
SLH_P_DATA (10) L2LF 233, 235
SLH_P_DATA (10) LGSF 200
SLH_P_HDR_LEN (4) L2LF 233, 235
SLH_P_HDR_LEN (4) LGSF 200
SLH_P_REC_LEN (0) L2LF 233, 235
SLH_P_REC_LEN (0) LGSF 200
SLH_P_REC_TYPE_FORK 4 L2LF 237
SLH_P_REC_TYPE_FORK 4 LGSF 203
SLH_P_REC_TYPE_NON_MOVED 4 L2LF 237
SLH_P_REC_TYPE_NON_MOVED 4 LGSF 203
SLH_P_REC_TYPE_NORMAL 4 L2LF 237
SLH_P_REC_TYPE_NORMAL 4 LGSF 203
SLH_P_REC_TYPE_SECONDARY 4 L2LF 237
SLH_P_REC_TYPE_SECONDARY 4 LGSF 203
SLH_P_REC_TYPE_TRIM 4 L2LF 237
SLH_P_REC_TYPE_TRIM 4 LGSF 203
SLH_P_REC_TYPE_USER 4 L2LF 237
SLH_P_REC_TYPE_USER 4 LGSF 203
SLH_P_STCK (8) L2LF 233, 235
SLH_P_STCK (8) LGSF 200
SLH_PREFIX 200
SLH_PREFIX (0) L2LF 235
SLH_REST (20) L2LF 235
SLH_REST (20) LGSF 200
SLH_SECONDARY (20) L2LF 236
SLH_SECONDARY (20) LGSF 201
SLH_TRIM (20) L2LF 236
SLH_TRIM (20) LGSF 201
SLH_USER (20) L2LF 236
SLH_USER (20) LGSF 201
SLO_READ 1 L2SL 241
SLO_RESTART 1 L2SL 241
SLO_WRITE 1 L2SL 241
SLOT (115) RMUW 339
SLOT (475) RMLK 306
SLOT (535) RMUW 339
SLOT (55) RMLK 305
SLR (0) TSAUX 388
SLR_CI_NUMBER (28) TSAUX 388
SLR_ITEM_NUMBER (20) TSAUX 388
SLR_LENGTH (0) TSAUX 388
SLR_NUMBER_OF_SECTIONS (24) TSAUX 388
SLR_PREV_OFFSET (2) TSAUX 388
SLR_QUEUE_NAME (8) TSAUX 388
SLR_RECORD_TYPE (4) TSAUX 388
SLR_SECTION_LENGTH (2A) TSAUX 388
SLR_SECTION_NUMBER (22) TSAUX 388
SLR_TIME_STAMP (18) TSAUX 388
SLR_TOTAL_LENGTH (26) TSAUX 388
sm
  sm macro-compatability anchor block, SMMCC 364
SM_ISOLATION_TOKEN (58) DSANC 54
SM_VARIABLE_SUBPOOL_TOKEN 58
SMA (0) SMDCC 345
SMA_ACTIVE_TASK_ALET_STEALS (1AC) SMDCC 347
SMA_ALET_COUNT (184) SMDCC 346
SMA_ALET_LIMIT (180) SMDCC 346
SMA_ARROW (2) SMDCC 345
SMA_BLOCK_NAME (8) SMDCC 345

SMA_CDSA_FIXED (BIT) SMDCC 345
SMA_COMMON_SS_CUMULATIVE_USERS 346
SMA_COMMON_SS_CURRENT_USERS (194) SMDCC 346
SMA_COMMON_SS_HWM_OF_USERS (198) SMDCC 346
SMA_COMMON_SUA_ADDRESS (16C) SMDCC 346
SMA_CTNFREEHEAD (134) SMDCC 346
SMA_CUMULATIVE_ALET_STEALS (1A8) SMDCC 347
SMA_DFH (3) SMDCC 345
SMA_DOMID (6) SMDCC 345
SMA_DSA_CURRENT_SIZE (12C) SMDCC 346
SMA_DSA_LIMIT (118) SMDCC 346
SMA_DSA_LIMIT_STORAGE (1B8) SMDCC 347
SMA_DSA_NON_EMPTY (138) SMDCC 346
SMA_DSAS_FIXED 345
SMA_DXHP (128) SMDCC 346
SMA_ECDSA_FIXED (BIT) SMDCC 345
SMA_EDSA_CURRENT_SIZE (130) SMDCC 346
SMA_EDSA_LIMIT (11C) SMDCC 346
SMA_EDSA_LIMIT_STORAGE (1BC) SMDCC 347
SMA_EDSA_NON_EMPTY (13C) SMDCC 346
SMA_ERDSA_FIXED (BIT) SMDCC 345
SMA_ESDSA_FIXED (BIT) SMDCC 345
SMA_EUDSA_FIXED (BIT) SMDCC 345
SMA_FLAGS (40) SMDCC 345
SMA_FLAGS2 (42) SMDCC 345
SMA_HWM_DSA_SIZE (1C0) SMDCC 347
SMA_HWM_EDSA_SIZE (1C4) SMDCC 347
SMA_ISOLATION_FLAGS (188) SMDCC 346
SMA_ISOLATION_STRUC (188) SMDCC 346
SMA_LAST_RESET_TIME (68) SMDCC 345
SMA_LAST_TUNING_TIME (1C8) SMDCC 347
SMA_LENGTH (0) SMDCC 345
SMA_LOC_EXPLICIT (BIT) SMDCC 345
SMA_MCAP (50) SMDCC 345
SMA_NOTIFIED_DSAS_NOT_CONSTRAINED (BIT) SMDCC 345
SMA_NUMBER_OF_SS_CREATES (1B0) SMDCC 347
SMA_NUMBER_OF_SS_DELETES (1B4) SMDCC 347
SMA_PPA_ABOVE_HEAD (88) SMDCC 346
SMA_PPA_BELOW_HEAD (84) SMDCC 346
SMA_PPA_FIRST 345
SMA_PPA_LAST (80) SMDCC 346
SMA_PPAP 346
SMA_PREFIX (0) SMDCC 345
SMA_PRIMARY_EXTENT_SIZE (90) SMDCC 346
SMA_QR_TCB (18C) SMDCC 346
SMA_RDSA_FIXED (BIT) SMDCC 345
SMA_REENTRANT_PROGRAM_PROTECT (BIT) SMDCC 345
SMA_SATP (110) SMDCC 346
SMA_SCA_DOMAIN_FIRST 345
SMA_SCA_DOMAIN_LAST (24) SMDCC 345
SMA_SCA_TASK_FIRST 345
SMA_SCA_TASK_LAST (1C) SMDCC 345
SMA_SCABLOCKHEAD (44) SMDCC 345
SMA_SCAFREEHEAD (14) SMDCC 345
SMA_SCANUM (28) SMDCC 345
SMA_SCQBLOCKHEAD (48) SMDCC 345
SMA_SCQFREEHEAD (10) SMDCC 345
SMA_SDSA_FIXED (BIT) SMDCC 345
SMA_SM_STATE (41) SMDCC 345
SMA_SMLOCK (3C) SMDCC 345
SMA_SMSY_RESUMED (BIT) SMDCC 345
SMA_SMX_COUNT (78) SMDCC 345
SMA_SMX_FIRST 345
SMA_SMX_LAST (38) SMDCC 345
SMA_SMXBLOCKHEAD (4C) SMDCC 345
SMA_SMXFREEHEAD (30) SMDCC 345
SMA_SOS_ABOVE (BIT) SMDCC 345
SMA_SOS_BELOW (BIT) SMDCC 345
SMA_SPIDNUM (2C) SMDCC 345
SMA_SQE_COUNT 345
SMA_SQE_FIRST (120) SMDCC 346
SMA_SQE_LAST (124) SMDCC 346
SMA_SQEBLOCKHEAD (54) SMDCC 345
SMA_SQEFREEHEAD (58) SMDCC 345
SMA_SQEHEAD (120) SMDCC 346
SMA_STATS_BUFFER_PTR (114) SMDCC 346
SMA_STORAGE_PROTECT (BIT) SMDCC 345
SMA_STORAGE_PROTECT_REQ (BIT) SMDCC 345
SMA_STORAGE_RECOVERY (BIT) SMDCC 345
SMA_SUA_ALL_POOLS_COUNT (172) SMDCC 346
SMA_SUA_ALLOC_FIRST 346
SMA_SUA_ALLOC_LAST (160) SMDCC 346
SMA_SUA_ALLOCATED_COUNT (17C) SMDCC 346
SMA_SUA_ARRAY_POOLHEAD (0) SMDCC 347
SMA_SUA_FREE_COUNT (170) SMDCC 346
SMA_SUA_FREEHEAD (148) SMDCC 346
SMA_SUA_POOL_AVG (17E) SMDCC 346
SMA_SUA_POOL_COUNT 346
SMA_SUA_POOL_FIRST 346
SMA_SUA_POOL_LAST (150) SMDCC 346
SMA_SUA_POOL_MIN (176) SMDCC 346
SMA_SUA_POOLHEAD (0) SMDCC 347
SMA_SUA_STEAL_FIRST 346
SMA_SUA_STEAL_LAST (168) SMDCC 346
SMA_SUABLOCKHEAD 346
SMA_SUSPENDED (10C) SMDCC 346
SMA_SYSTEM_SUSPEND_TOKEN (64) SMDCC 345
SMA_SYSTEM_TASK_NOTIFIES (60) SMDCC 345
SMA_SYSTEM_TASK_RUNS (5C) SMDCC 345
SMA_TRANSACTION_ISOLATION (BIT) SMDCC 346
SMA_TRANSACTION_ISOLATION_REQ (BIT) SMDCC 345
SMA_UDSA_FIXED (BIT) SMDCC 345
SMA_UNIQUE_SS_CUMULATIVE_USERS (19C) SMDCC 347
SMA_UNIQUE_SS_CURRENT_USERS (1A0) SMDCC 347
SMA_UNIQUE_SS_HWM_OF_USERS (1A4) SMDCC 347
SMDCC 345
SMF_BLOCK_HEADER (60) L2BL 210
SMF_DATA_SECTION 210, 232
SMF_EMPTY (BIT) STUCB 376
SMF_HEADER (0) L2LF 232
SMF_HEADER (60) L2BL 210
SMF_MAX_BLOCK_LEN 4 L2LF 237
SMF_MAX_DATA_SECTION_LEN 4 L2LF 237
SMF_PRODUCT_SECTION 210, 232
SMF_PTR (50) STCB1 374
SMF_REASON (1B8) L2BS 216
SMF_REASON (1B8) L2SR 248
SMF_REASON (C8) L2HS 230
SMF_REC_INDEX (8C8) STUCB 376
SMF_REC_PTR (8C4) STUCB 376
SMF_RECORD_COUNT (8E8) STUCB 376
SMF_RESPONSE (1B4) L2BS 216
SMF_RESPONSE (1B4) L2SR 248
SMF_RESPONSE (C4) L2HS 230
SMFDS_DATA (9E) L2LF 232
SMFDS_DATA (FE) L2BL 210
SMFH_APS (1C) L2LF 232
SMFH_APS (7C) L2BL 210
SMFH_ASL (28) L2LF 232
SMFH_ASL (88) L2BL 210
SMFH_ASN (2A) L2LF 232
SMFH_ASN (8A) L2BL 210
SMFH_ASS (24) L2LF 232
SMFH_ASS (84) L2BL 210
SMFH_DATA_SECT_LENGTH 4 L2LF 237
SMFH_DATA_SECT_NUMBER 4 L2LF 237
SMFH_DATA_SECT_OFFSET 4 L2LF 237
SMFH_DTE (6A) L2BL 210
SMFH_DTE (A) L2LF 232
SMFH_FLG (4) L2LF 232
SMFH_FLG (64) L2BL 210
SMFH_FLG_ESA4 1 L2LF 237
SMFH_LEN (0) L2LF 232
SMFH_LEN (60) L2BL 210
SMFH_LPS (20) L2LF 232
SMFH_LPS (80) L2BL 210
SMFH_MFL_ID 4 L2LF 237
SMFH_NPS (22) L2LF 232
SMFH_NPS (82) L2BL 210
SMFH_NUMBER_TRIPLETS 4 L2LF 237
SMFH_PRD_SECT_LENGTH 4 L2LF 237
SMFH_PRD_SECT_NUMBER 4 L2LF 237
SMFH_PRD_SECT_OFFSET 4 L2LF 237
SMFH_RSVD1 (1A) L2LF 232
SMFH_RSVD1 (7A) L2BL 210
SMFH_RTY (5) L2LF 232
SMFH_RTY (65) L2BL 210
SMFH_RTY_110 1 L2LF 237
SMFH_SEG (2) L2LF 232
SMFH_SEG (62) L2BL 210
SMFH_SID (6E) L2BL 210
SMFH_SID (E) L2LF 232
SMFH_SSI (12) L2LF 232
SMFH_SSI (72) L2BL 210
SMFH_SSI_CICS 4 L2LF 237
SMFH_STY (16) L2LF 232

SMFH_STY (76) L2BL 210
SMFH_STY_LG 2 L2LF 237
SMFH_STY_MN 2 L2LF 237
SMFH_STY_ST 2 L2LF 237
SMFH_TME (6) L2LF 232
SMFH_TME (66) L2BL 210
SMFH_TRN (18) L2LF 232
SMFH_TRN (78) L2BL 210
SMFLOGBLOCKHEADER (0) L2LF 232
SMFPS_JBN (7E) L2LF 232
SMFPS_JBN (DE) L2BL 210
SMFPS_JNM (76) L2LF 232
SMFPS_JNM (D6) L2BL 210
SMFPS_MFL (3E) L2LF 232
SMFPS_MFL (9E) L2BL 210
SMFPS_MFL_0 2 L2LF 237
SMFPS_PDN (96) L2LF 232
SMFPS_PDN (F6) L2BL 210
SMFPS_PRN (2E) L2LF 232
SMFPS_PRN (8E) L2BL 210
SMFPS_RSD (86) L2LF 232
SMFPS_RSD (E6) L2BL 210
SMFPS_RST (8A) L2LF 232
SMFPS_RST (EA) L2BL 210
SMFPS_RSVD2 (40) L2LF 232
SMFPS_RSVD2 (A0) L2BL 210
SMFPS_RSVD3 (42) L2LF 232
SMFPS_RSVD3 (A2) L2BL 210
SMFPS_SPN (36) L2LF 232
SMFPS_SPN (96) L2BL 210
SMFPS_UIF (8E) L2LF 232
SMFPS_UIF (EE) L2BL 210
SMFPS_VRM (2C) L2LF 232
SMFPS_VRM (8C) L2BL 210
SMFPS_VRM_VAL 2 L2LF 237
SMLOCK_NAME 8 SMDCC 356
SMMCC 364
SMX (0) SMDCC 350
SMX_CICS_DATAKEY (BIT) SMDCC 350
SMX_CICS24_P (20) SMDCC 350
SMX_CICS31_P (24) SMDCC 350
SMX_CLEAR_STG (BIT) SMDCC 350
SMX_EYECATCHER (0) SMDCC 350
SMX_FLAGS (10) SMDCC 350
SMX_FREEZE_STG (BIT) SMDCC 350
SMX_ISOLATE (BIT) SMDCC 350
SMX_NAME 4 SMDCC 356
SMX_NEXT (4) SMDCC 350
SMX_PREFIX (0) SMDCC 350
SMX_PREV (8) SMDCC 350
SMX_REMOTE_TRAN (BIT) SMDCC 350
SMX_SUBPOOL_TOKEN_TABLE (20) SMDCC 350
SMX_SUBSPACE_ACTIVE (BIT) SMDCC 350
SMX_SUBSPACE_TASK (BIT) SMDCC 350
SMX_SUBSPACE_TOKEN (C) SMDCC 350
SMX_TASKDATALOC_ANY (BIT) SMDCC 350
SMX_TRANSACTION_NUMBER 350
SMX_TRANSACTION_TOKEN (18) SMDCC 350
SMX_USER24_P (28) SMDCC 350
SMX_USER31_P (2C) SMDCC 350
SMXBLOCK_NAME 8 SMDCC 356
SMXBLOCK_SIZE 4 SMDCC 363
SO_LISTENER_STATE_CLOSED 1 SOA 372
SO_LISTENER_STATE_CLOSING 1 SOA 372
SO_LISTENER_STATE_IMMCLOSING 1 SOA 372
SO_LISTENER_STATE_OPEN 1 SOA 372
SO_LISTENER_STATE_OPENING 1 SOA 372
SO_OFTEN_CE (40) DSANC 54
SO_OFTEN_SHP (28) DSANC 54
SO_SERVICE_WLM_STATE_AVAILABLE 1 SOA 372
SO_SERVICE_WLM_STATE_DEREGERROR 1 SOA 372
SO_SERVICE_WLM_STATE_DEREGISTERED 1 SOA 372
SO_SERVICE_WLM_STATE_NOTAPPLIC 1 SOA 372
SO_SERVICE_WLM_STATE_REGERROR 1 SOA 372
SO_SERVICE_WLM_STATE_REGISTERED 1 SOA 372
SO_SERVICE_WLM_STATE_UNAVAILABLE 1 SOA 372
SO_SERVICE_WLM_STATE_UNREGISTERED 1 SOA 372
SO_STATE_INITIALISED 1 SOA 372
SO_STATE_INITIALISING 1 SOA 372
SO_STATE_QUIESCED 1 SOA 372
SO_STATE_QUIESCING 1 SOA 372
SO_STATE_TERMINATED 1 SOA 372
SOA 366
SOA (0) SOA 366
SOA_ARROW (2) SOA 366
SOA_BLOCK_NAME (8) SOA 366
SOA_CEEPIPI_ENTRY (68) SOA 367
SOA_COLD_START (BIT) SOA 366
SOA_DFH (3) SOA 366
SOA_DFHSOSE_ENTRY (64) SOA 367
SOA_DFHSOSE_SUFFIX (E0) SOA 367
SOA_DOMID (6) SOA 366
SOA_FLAGS1 (12) SOA 366
SOA_FLAGS2 366
SOA_FLAGS3 (14) SOA 366
SOA_GENERAL_SPTOKEN (3C) SOA 367
SOA_GSK (98) SOA 367
SOA_KEYFILE_PASSWORD (C8) SOA 367
SOA_KEYFILE_PATHNAME (98) SOA 367
SOA_LAST_RESET_TIME (F0) SOA 367
SOA_LENGTH (0) SOA 366
SOA_LISTENER_STATE (11) SOA 366
SOA_LOCK_TOKEN (1C) SOA 366
SOA_LTE_CHAIN (70) SOA 367
SOA_LTE_EMPTY_ECB (74) SOA 367
SOA_LTE_HEAD (78) SOA 367
SOA_LTE_NUM_ENTRIES (70) SOA 367
SOA_LTE_SPTOKEN (44) SOA 367
SOA_NAMESERVER_ERR (BIT) SOA 366
SOA_NORMAL_ENCRYPTION (BIT) SOA 366
SOA_PREFIX (0) SOA 366
SOA_SELECT_WAIT (BIT) SOA 366
SOA_SELECTEX_ECB (18) SOA 366
SOA_SL_MODENAME_TOKEN (2C) SOA 367
SOA_SL_TCB_TOKEN (5C) SOA 367
SOA_SO_MODENAME_TOKEN (20) SOA 367
SOA_SO_STATE (10) SOA 366
SOA_SO_STOKEN (34) SOA 367
SOA_SO_TCB_TOKEN (54) SOA 367
SOA_SOIS_CEEPIPI_TOKEN (30) SOA 367
SOA_SSL_AVAILABLE (BIT) SOA 366
SOA_SSL_SUBTASKS 367
SOA_SSLV2_TIMEOUT (D8) SOA 367
SOA_SSLV3_TIMEOUT (DC) SOA 367
SOA_STATS_BUFFER_PTR (F8) SOA 367
SOA_STE_SPTOKEN (4C) SOA 367
SOA_STRONG_ENCRYPTION 366
SOA_TCBPOOL_LOCK_TOKEN (28) SOA 367
SOA_TCPIP_REQUIRED (BIT) SOA 366
SOA_TCPIPSERVICE_CLASSP (E8) SOA 367
SOA_TCPIPSERVICE_LOCK_TOKEN (24) SOA 367
SOA_TOKEN_COUNTER (EC) SOA 367
SOA_WLM_DATA (FC) SOA 367
SOA_WLM_HOSTNAME (108) SOA 367
SOA_WLM_SERVERNAME 367
SOA_WLM_STATE (FC) SOA 367
SOA_XRSINDI_ACTIVE (BIT) SOA 366
SOCK_DATA (1B0) SOA 369
SOCK_DATA (8A) SOA 370
SOCK_FAMILY (1AF) SOA 369
SOCK_FAMILY (89) SOA 370
SOCK_LEN (1AE) SOA 369
SOCK_LEN (88) SOA 370
SOCK_SIN_ADDR (1B2) SOA 369
SOCK_SIN_ADDR (8C) SOA 370
SOCK_SIN_PORT (1B0) SOA 369
SOCK_SIN_PORT (8A) SOA 370
SOCK_SUN_NAME (1B0) SOA 369
SOCK_SUN_NAME (8A) SOA 370
SOCKADDR_ADDR (20) SOA 371, 372
SOCKADDR_LENGTH (1C) SOA 371, 372
SOCKET_DESCRIPTOR (18) SOA 371, 372
SOCKET_ID (20) SOA 372
SOCKET_PARMS (18) SOA 371
SOCKET_VECTOR (28) SOA 371
sockets
  sockets anchor block, SOA 366
SOLITAIRE_SYSTEM_LOG (0) RMSL 329
SOR_CICS_INFO (0) L2LF 233
SOR_CICS_INFO (FE) L2BL 210
SOR_CICS_RELEASE (0) L2LF 233
SOR_CICS_RELEASE (FE) L2BL 210
SOR_CICS_USERNAME (10A) L2BL 211
SOR_CICS_USERNAME (C) L2LF 233
SOR_DATA (FE) L2BL 210
SOR_REC_TYPE 2 L2LF 237

SOR_SPECIFIC_APPLID (102) L2BL 211
SOR_SPECIFIC_APPLID (4) L2LF 233
SORT_RECORD_LEN (8C0) STUCB 376
SORT_RECORD_PTR (8BC) STUCB 376
SOURCE_REF (78) BAACT 7, 8, 13, 19
SPC (0) SMDCC 352
SPC_TUNING_AVERAGE (4) SMDCC 352
SPC_TUNING_INTERVALS (0) SMDCC 352
SPC_TYPE 8 SMDCC 356
SPECIAL_APPLID 1 MEMMS 256
SPECIAL_AREA (A0) DSANC 55
SPECIAL_DATE 1 MEMMS 256
SPECIAL_INSERT_ELEMENT 1 MEMMS 256
SPECIAL_NETNAME 1 MEMMS 256
SPECIAL_PRIMAB 1 MEMMS 256
SPECIAL_PROGNAME 1 MEMMS 256
SPECIAL_SECAB 1 MEMMS 256
SPECIAL_SYSID 1 MEMMS 256
SPECIAL_TERMID 1 MEMMS 256
SPECIAL_TIME 1 MEMMS 256
SPECIAL_TRANID 1 MEMMS 256
SPECIAL_TRANNUM 1 MEMMS 256
SPECIAL_TYPE (BIT) DSTSK 66
SPECIAL_TYPE_IMMEDIATE_SHUTDOWN (BIT) DSTSK 66
SPECIAL_TYPE_SMSY (BIT) DSTSK 66
SPECIAL_USERID 1 MEMMS 256
SPID_DOMAIN_FIRST 4 SMDCC 356
SPID_FREE 4 SMDCC 356
SPID_TASK_CICS24 4 SMDCC 356
SPID_TASK_CICS31 4 SMDCC 356
SPID_TASK_USER24 4 SMDCC 356
SPID_TASK_USER31 4 SMDCC 356
SPNAME_CONTROL 8 SMMCC 366
SPNAME_GENERAL 8 DHANC 42
SPNAME_GENERAL 8 LGANC 193
SPNAME_GENERAL 8 USANC 409
SPNAME_GENERAL 8 XSANC 450
SPNAME_SHARED 8 SMMCC 366
SPNAME_SHRC24 8 SMMCC 366
SPNAME_SHRC31 8 SMMCC 366
SPNAME_SHRU24 8 SMMCC 366
SPNAME_SHRU31 8 SMMCC 366
SPNAME_TP 8 SMMCC 366
SPNAME_TP24 8 SMMCC 366
SQE (0) SMDCC 354
SQE_BYTES_REQUESTED (C) SMDCC 354
SQE_DELETED (BIT) SMDCC 354
SQE_FLAGS (28) SMDCC 354
SQE_NEXT (0) SMDCC 354
SQE_PREV (4) SMDCC 354
SQE_SCAP (8) SMDCC 354
SQE_SUSPEND_START (18) SMDCC 354
SQE_SUSPEND_TOKEN (10) SMDCC 354
SQE_TASK_TOKEN (14) SMDCC 354
SQE_TRANSACTION_NUMBER 354
SQEBLOCK_NAME 8 SMDCC 356
SQEBLOCK_SIZE 4 SMDCC 363
SR_FIXED_STORAGE 4 MEPS 259
SR_PRIMLEN 4 MEPS 259
SR_SECLEN 4 MEPS 259
SR_TOTAL_LEN 4 MEPS 259
SR_VARLEN 4 MEPS 259
SSC_INIT 47
SSC_QUIESCE (BIT) DMCB1 47
SSC_TERM (BIT) DMCB1 47
SSL_SUBTASK_VECTOR (0) SOA 370
SSLT_ACTIVE_TCBS (12) SOA 371
SSLT_ARROW (2) SOA 371
SSLT_BLOCK_NAME (8) SOA 371
SSLT_BUSY (BIT) SOA 371
SSLT_CEEPIPI_TOKEN (18) SOA 371
SSLT_DFH (3) SOA 371
SSLT_DOMID (6) SOA 371
SSLT_FLAG1 (0) SOA 371
SSLT_FLAG2 (1) SOA 371
SSLT_INIT_FAILED (BIT) SOA 371
SSLT_INIT_STARTED 371
SSLT_INITIALIZED 371
SSLT_LENGTH (0) SOA 370
SSLT_MAX_TCBS (10) SOA 371
SSLT_MODE_TOKEN (14) SOA 371
SSLT_PREFIX (0) SOA 370
SSLT_STE_ADDRESS (8) SOA 371
SSLT_TCB_ADDRESS (C) SOA 371
SSLT_TCB_COUNTERS (10) SOA 371
SSLT_TCB_ENTRY (18) SOA 371
SSLT_TCB_TOKEN (10) SOA 371
SSLTCB_ENTRY (0) SOA 371
STA_BROWSES (194) LDCBS 170
STA_DEB_REBUILDS (1CC) LDCBS 170
STA_DEFINES (184) LDCBS 170
STA_DELETES (188) LDCBS 170
STA_FETCH_TIME (1B0) LDCBS 170
STA_FETCHS 170
STA_INQUIRES (18C) LDCBS 170
STA_LAST_RESET_TIME (1D0) LDCBS 170
STA_LONGEST_NAME (1A0) LDCBS 170
STA_NAME_ADDED (1A4) LDCBS 170
STA_NAME2LONG (19C) LDCBS 170
STA_NOTIFIES (198) LDCBS 170
STA_REFRESHS (190) LDCBS 170
STA_TIMES_WAITS_HWM (1C4) LDCBS 170
STA_USES (1B4) LDCBS 170
STA_WAIT_TIME (1C8) LDCBS 170
STA_WAITS (1B8) LDCBS 170
STA_WAITS_HWM (1C0) LDCBS 170
STA_WAITS_PAST (1BC) LDCBS 170
stack
    kernel stack entry, KESTP 163
    stack segment table header, LIFO 203
 STAFB 373
STANDARD_PASS 1 STUCB 378
STANDBY 1 PAA 284
START (0) L2BL 209
START (38) L2BL 208
START (C8) L2CH 223
START_ALL (BIT) PAA 283
START_DELIVERY (18) RMLI 304
START_DELIVERY (80) RMUW 338
START_DELIVERY (8C0) RMLK 306
START_HIGH (24E) L2BS 217
START_HIGH (24E) L2SR 249
START_OF_MESSAGE 1 MEMMS 256
START_SPECIFIED (13) PAA 283
START_TIME (250) L2BS 217
START_TIME (250) L2SR 249
START_WRITE_COMPLETE 1 L2SR 251
START_WRITE_ISSUED 1 L2SR 251
STARTED (8C) BAACT 17
STARTED (AC) BAACT 11
STARTOFRUNDATA (0) L2LF 233
STASK (8) DSTSK 64, 67
state
    dm authorised facility state, DMAFC 45
    domain manager enf state, DMENC 52
    web state manager data, WBSTC 422
STATE (28) DSTSK 64, 67
STATE_AFTER_COMMIT (CC) CPCPS 33
STATE_CHANGE_TIME (60) RMLK 312
STATE_CHANGE_TIME (60) RMUW 331
STATE_LOCK_NAME 8 LDCBS 175
STATE_LOCK_NAME 8 MNCBS 273
static
    catalog static storage, CCGD 29
    cics/db2 static storage, D2SS 96
    cpi static storage area, CPSPS 34
    partner domain static storage area, PRS 296
statistics
    dce services domain global statistics, DEGPC 38
    statistics authorised parameter block, STAFB 373
    statistics domain anchor block, STCB1 374
    statistics utility program anchor block, STUCB 375
    user domain statistics, USGPS 409
STATISTICS 8 LDCBS 175
STATISTICS_PTR (58) STCB1 374
STATS_APPLID (0) STUCB 377
STATS_BUFFER_PTR (5E8) DSANC 57
STATS_BUFFER_SIZE 4 SMDCC 362
STATS_COLL_TYPE (838) STUCB 375
STATS_DATES (20) STUCB 377
STATS_EODES (C) STUCB 377
STATS_FILE_OPEN (8E5) STUCB 376
STATS_INTERVALS (8) STUCB 377
STATS_INTES (10) STUCB 377
STATS_OK (C4) L2BS 213
STATS_OK (C4) L2SR 244

STATS_RECORD_COUNT (8F0) STUCB 376
STATS_REQES (14) STUCB 377
STATS_RRTES (18) STUCB 377
STATS_SELECTED_COUNT (8F4) STUCB 376
STATS_TIMES (30) STUCB 377
STATS_USSES (1C) STUCB 377
STATUS (1A) BAPT 23
STATUS (2C) RMLK 311
STATUS (2C) RMUW 330
STATUS (38) L2SR 250
STATUS (70) L2BS 213
STATUS (70) L2SR 244
STATUS (B0) L2BS 213
STATUS (B0) L2SR 244
STATUS_FLAGS 376
STATUS_LOG_RECORD 4 RMUW 336, 340
STCB1 374
STCK_TYPE (0) FCQSE 106
STCK_VALUE (0) L2HP 226
STCK_VALUE (58) L2CH 221
STCK_VALUE (90) L2CH 223
STE 399
STE (0) SOA 369
STE_ADDR (8A) SOA 370
STE_ARROW (2) SOA 369
STE_BLOCK_NAME (8) SOA 369
STE_CID (1F8) SOA 370
STE_CLIENT_IP_ADDRESS (68) SOA 370
STE_CS_FLAG_BYTE1 (20) SOA 369
STE_CS_FLAG_WORD (20) SOA 369
STE_DFH (3) SOA 369
STE_DOMID (6) SOA 369
STE_ERROR_CODE (50) SOA 370
STE_ERROR_DATA (50) SOA 370
STE_ERROR_FORMAT (54) SOA 370
STE_ERROR_FUNCTION 370
STE_ERROR_REASON (64) SOA 370
STE_ERROR_RESPONSE (60) SOA 370
STE_ERROR_TRANID (58) SOA 370
STE_ERROR_TRANNUM (5C) SOA 370
STE_FLAG1 (1C) SOA 369
STE_FLAG2 369
STE_IDENTITY_NO (34) SOA 370
STE_INET_ADDR (8A) SOA 370
STE_LENGTH (0) SOA 369
STE_NEXT (0) TSRL 399
STE_NEXT (10) SOA 369
STE_PCAP (C) TSRL 399
STE_PREFIX (0) SOA 369
STE_PREFIX (0) TSRL 399
STE_PREV (14) SOA 369
STE_PREV (4) TSRL 399
STE_RAIOCB (178) SOA 370
STE_RECV_ASYNC_ECB (3C) SOA 370
STE_RECV_IN_PROGRESS (BIT) SOA 370
STE_RECV_TIMEOUT (38) SOA 370
STE_REF_COUNT (28) SOA 370
STE_REPOSITORY_TOKEN (80) SOA 370
STE_SAIOCB (F8) SOA 370
STE_SEND_ASYNC_ECB (40) SOA 370
STE_SEND_IN_PROGRESS (BIT) SOA 370
STE_SERVICE_LTE_ID (30) SOA 370
STE_SERVICE_LTE_PTR (2C) SOA 370
STE_SERVICE_LTE_TOKEN (2C) SOA 370
STE_SESSION_ERROR (BIT) SOA 369
STE_SOCKADDR (88) SOA 370
STE_SOCKADDR_HEADER (88) SOA 370
STE_SOCKET (18) SOA 369
STE_SOCKET_CLOSED (BIT) SOA 369
STE_SOCKET_GIVEN (BIT) SOA 369
STE_SOCKET_TAKEN (BIT) SOA 369
STE_SSL_COMPLETE (BIT) SOA 369
STE_SSL_HANDLE (44) SOA 370
STE_SSL_LE_TOKEN (4C) SOA 370
STE_SSL_REQUIRED (BIT) SOA 369
STE_SSL_THREAD_PTR (48) SOA 370
STE_SYSID (8) TSRL 399
STE_TERMINATION (BIT) SOA 369
STE_TXN_COUNT (24) SOA 370
STE_UNIX_ADDR 370
STE_USERID (78) SOA 370
STE_USERID_LEN (77) SOA 370
STE_USERID_STRUCTURE (77) SOA 370
STGTYPE (0) TSQU 398
STGTYPE_AUX_TST 1 TSQU 398
STGTYPE_MAIN 1 TSQU 398
STIMER_ARRAY (20) DSANC 61
STIMER_BLOCK (0) DSANC 60
STIMER_BLOCK_ADDR (34) DSANC 61
STIMER_BLOCK_PTR (40) DSANC 59
STIMER_DSTCB (1C) DSANC 61
STIMER_ENTRY_ADDR (30) DSANC 61
STIMER_FLAGS (18) DSANC 61
STIMER_RUN (BIT) DSANC 61
STIMER_TIME (24) DSANC 61
STIMER_TOKEN (2C) DSANC 61
storage
    catalog static storage, CCGD 29
    cics/db2 static storage, D2SS 96
    cpi static storage area, CPSPS 34
    partner domain static storage area, PRS 296
    security supervisor storage, XSSS 451
    storage manager anchor block, SMDCC 345
    temporary storage anchor block, TSA 380
    temporary storage auxiliary class, TSAUX 384
    temporary storage main class, TSMN 392
    temporary storage model class, TSMN 390
    temporary storage name class, TSNM 393
    temporary storage ownership lock class, TSOL 394
    temporary storage queue class, TSQU 396
    temporary storage resource lock class, TSRL 401
    temporary storage shared class, TSRL 399
    temporary storage wait queue class, TSWQ 402
STORAGE_NOTIFY 8 LDCBS 175
STORAGE_SHORTFALL (78) DSANC 54
STORAGE_VIOLATION_DATA_LEN 4 SMDCC 356
STORE_CRITICAL_POINT (60) DSANC 54
STORE_POINTER (28) RMLK 316
STORE_POINTER (98) RMNM 322
STORE_SHORT_POINT (5C) DSANC 54
STORECLOCK (0) CAUTR 27
STQ_CONVID (5C) FEP06 123
STQ_DATALENGTH (28) FEP06 123
STQ_DATATYPE 123
STQ_DEVICE (64) FEP06 123
STQ_EVENT1 (38) FEP06 123
STQ_EVENT2 (3C) FEP06 123
STQ_EVENTDATA (38) FEP06 123
STQ_EVENTTYPE (30) FEP06 123
STQ_EVENTVALUE (34) FEP06 123
STQ_FLENGTH (74) FEP06 123
STQ_FORMAT (68) FEP06 123
STQ_NODE (54) FEP06 123
STQ_POOL (44) FEP06 123
STQ_QUEUER (24) FEP06 123
STQ_SPARE4 (40) FEP06 123
STQ_SPARE8 (6C) FEP06 123
STQ_TARGET (4C) FEP06 123
STQ_TERMID (FC) FEP06 123
STQ_TRANSID (F8) FEP06 123
STQ_USERDATA (78) FEP06 123
STQDATA (28) FEP06 123
stream
    log manager browseable stream class, L2BS 211
    log manager hard stream class, L2HS 227
    log manager stream class, L2SR 242
STREAM (0) L2SR 242
STREAM_CHAIN_LINK (18) L2BS 212
STREAM_CHAIN_LINK (18) L2SR 243
STREAM_FACTORY (38) L2SR 249
STREAM_FORCE_TOKEN (28) L2BS 212
STREAM_FORCE_TOKEN (28) L2SR 243
STREAM_INSTANCE_DATA 212, 243
STREAM_JOURNAL (C8) L2BS 213
STREAM_JOURNAL (C8) L2SR 244
STREAM_RESOURCES (40) L2CH 221
STREAMBLOCK (0) L2SR 250
STRING_BROWSE_RC (6E) CCGD 30
STRING_BUFFER (0) CCGD 30
STRING_BUFFER_A (5C) CCGD 30
STRING_BUFFER_DATA (1C) CCGD 31
STRING_BUFFER_DOM (0) CCGD 31
STRING_BUFFER_DOM_TYPE (0) CCGD 30
STRING_BUFFER_KEY (0) CCGD 30
STRING_BUFFER_NAME (C) CCGD 31
STRING_BUFFER_TYPE (4) CCGD 31

STRING_DOM (70) CCGD 30
STRING_DOM_TYPE (70) CCGD 30
STRING_EYECATCHER (50) CCGD 30
STRING_FUNCTION 30
STRING_KEY (70) CCGD 30
STRING_NAME (7C) CCGD 30
STRING_RPL_A (58) CCGD 30
STRING_RPL_FEEDBACK (8D) CCGD 30
STRING_STATES (6C) CCGD 30
STRING_STORAGE (50) CCGD 29
STRING_TASKNUM (94) CCGD 30
STRING_TOKEN (64) CCGD 30
STRING_TRANSID (90) CCGD 30
STRING_TYPE (74) CCGD 30
STRING_VSAM_DEBUG (8C) CCGD 30
STRING_VSAM_RECORD_A (60) CCGD 30
STRING_VSAM_REQUEST (8C) CCGD 30
STRING_XC (BIT) CCGD 30
STRING_XC_WAIT_ECB (68) CCGD 30
STRUCTURE_NAME (12E) L2BS 216
STRUCTURE_NAME (12E) L2SR 248
STRUCTURE_NAME (3E) L2HS 230
structures
  directory manager structures, DDCBC 36
STUCB 375
STUP_APPLID_STATS (0) STUCB 377
STUP_KERNEL_PTR (8CC) STUCB 376
STYPE (4D) L2BL 208
SUA (0) SMDCC 352
SUA_ALLOCATED_TO_TASK (BIT) SMDCC 352
SUA_EYECATCHER (0) SMDCC 352
SUA_FLAGS (34) SMDCC 352
SUA_NAME 4 SMDCC 356
SUA_NEXT (4) SMDCC 352
SUA_OPEN_ALET (18) SMDCC 352
SUA_POOL_INDEX (30) SMDCC 352
SUA_POOL_OR_ALLOC_CHAIN (4) SMDCC 352
SUA_PREFIX (0) SMDCC 352
SUA_PREV (8) SMDCC 352
SUA_QR_ALET (14) SMDCC 352
SUA_STEAL_NEXT (C) SMDCC 352
SUA_STEAL_PREV (10) SMDCC 352
SUA_STOKEN (1C) SMDCC 352
SUA_SUBSPACE_NAME (24) SMDCC 352
SUA_TASK_TOKEN (2C) SMDCC 352
SUABLOCK_NAME 8 SMDCC 356
SUABLOCK_SIZE 4 SMDCC 363
SUB_DISP 57
SUB_DISPATCHER (0) DSANC 60
SUB_GEN_NO (3C) BAACT 14
SUB_MODE (40) BAACT 14
SUBD_FLAGS (1A0) DSANC 57
SUBD_FLAGS (20) DSANC 60
SUBD_MODE (1A4) DSANC 57
SUBD_MODE (24) DSANC 60
SUBD_MODENAME (1A8) DSANC 57
SUBD_MODENAME (28) DSANC 60
SUBPOOL_NAME (20) BAACT 17
SUBPOOL_NAME (20) L2BL 209
SUBPOOL_NAME (48) L2BS 219
SUBPOOL_NAME (48) L2CH 222
SUBPOOL_NAME (48) L2SR 249
SUBPOOL_NAME (50) RMUW 337
SUBPOOL_NAME (890) RMLK 306
SUBPOOL_NAME_PREFIX (20) BAACT 17
SUBPOOL_NAME_PREFIX (20) L2BL 209
SUBPOOL_NAME_PREFIX (48) L2BS 219
SUBPOOL_NAME_PREFIX (48) L2CH 223
SUBPOOL_NAME_PREFIX (48) L2SR 250
SUBPOOL_NAME_PREFIX (50) RMUW 338
SUBPOOL_NAME_PREFIX (890) RMLK 306
SUBPOOL_NAME_SUFFIX (24) BAACT 17
SUBPOOL_NAME_SUFFIX (24) L2BL 210
SUBPOOL_NAME_SUFFIX (4C) L2BS 219
SUBPOOL_NAME_SUFFIX (4C) L2CH 223
SUBPOOL_NAME_SUFFIX (4C) L2SR 250
SUBPOOL_NAME_SUFFIX (54) RMUW 338
SUBPOOL_NAME_SUFFIX (894) RMLK 306
SUBPOOL_TOKEN (14) PRS 296
SUBPOOL_TOKEN (28) BAACT 17
SUBPOOL_TOKEN (28) L2BL 210
SUBPOOL_TOKEN (40) STCB1 374
SUBPOOL_TOKEN (50) L2BS 219
SUBPOOL_TOKEN (50) L2CH 223
SUBPOOL_TOKEN (50) L2SR 250
SUBPOOL_TOKEN (58) RMUW 338
SUBPOOL_TOKEN (898) RMLK 306
SUBPTOK (0) DMCB3 50
SUBPTOK (794) DMCB1 47
SUBPTOK_N (4) DMCB3 50
SUBPTOK_N (798) DMCB1 47
SUBPTOK_P (0) DMCB3 50
SUBPTOK_P (794) DMCB1 47
SUBSPACE_ELIGIBLE (BIT) DSANC 59
SUBSPACE_TOKEN (80) DSANC 59
SUM (BIT) STUCB 376
SUM_TOT_REC_LENGTH (8B8) STUCB 376
SUM_TOT_REC_PTR (8B4) STUCB 376
SUMMARY_PASS 1 STUCB 378
SUMMARY_REC_LENGTH (8A8) STUCB 376
SUMMARY_REC_PTR (8A4) STUCB 376
supervisor
  security supervisor storage, XSSS 451
support
  device support extension, FEP08 127
SURVIVED_COLD_START (BIT) RMLK 312
SURVIVED_COLD_START (BIT) RMUW 331
SUSPEND 1 DSTSK 67
SUSPEND_CELL_ROOT (D0) DSANC 55
SUSPEND_CS_WORD 64, 67
SUSPEND_FTOKEN (18) L2SR 251
SUSPEND_PAGE_MAP (10) DSANC 62
SUSPEND_QUEUE (10) L2SR 250
SUSPEND_QUEUE (48) L2BS 212
SUSPEND_QUEUE (48) L2SR 244
SUSPEND_QUEUE (88) L2BS 213
SUSPEND_QUEUE (88) L2SR 244
SUSPEND_RESUME_AREA (0) DSTSK 67
SUSPEND_RESUME_AREAS_IN_BLOCK 4 DSTSK 68
SUSPEND_STATUS (1C) L2SR 251
SUSPEND_TOKEN (10) L2SR 251
SUSPEND_TOKEN (90) RMLK 312
SUSPEND_TOKEN (90) RMUW 332
SUSPENDED_AWAITING_OPEN_TCB (65C) DSANC 58
SUSPENDELEMENT 251
SVC
  data tables SVC routine anchor blocks, DTSPS 72
SWITCH_PARMS (158) XCCBC 433
SYMPTOM_INSERT 1 MEMMS 256
SYMPTOM_RECORD (0) MEPS 258
SYMPTOM_RECORD_CHAR (0) MEPS 258
SYMPTOM_SPECIAL 1 MEMMS 256
SYMPTOM_TEXT 1 MEMMS 256
SYMSTRING_DEF 1 MEMMS 256
SYNC_LEVEL (70) CPCPS 33
SYNCPOINT_RETURN_CODE (D0) CPCPS 33
SYSIN_EOF (BIT) PAA 283
SYSIN_FIRST_RECORD (BIT) PAA 283
SYSIN_FLAG (BIT) PAA 283
SYSIN_POINTERS (40) PAA 283
SYSIN_RECORD_L 2 PAA 284
SYSIN_SAVED (BIT) PAA 283
SYSIN_STATUS (BIT) PAA 283
SYSLOG (4C) L2BL 208
SYSLOG (C6) L2BS 213
SYSLOG (C6) L2SR 244
SYSLOGBLOCKHEADER (0) L2LF 232
SYSLOGCOMBINEDRECORD (0) L2LF 235
SYSLOGFAILURE 241
SYSLOGOPERATION (0) L2SL 241
SYSLOGRECORD (0) L2LF 233
SYSLOGUSER (0) L2LF 236
system
  log manager system log class, L2SL 240
  recovery manager system log class data, RMSL 329
  recovery manager system log instance, RMSL 327
  system log format, LGSF 199
SYSTEM 1 DSTSK 67
SYSTEM_LOG (12C) L2BS 216
SYSTEM_LOG (12C) L2SR 248
SYSTEM_LOG (3C) L2HS 230
SYSTEM_LOG_CHAIN_TOKEN 312, 331
SYSTEM_LOG_REGISTER 327, 329
SYSTEM_RESTART_STATES (1D0) RMLK 315
SYSTEM_RESTART_STATES (1D0) RMUW 334
SYSTEM_RESTART_STATES (20) RMRO 325

SYSTEM_STATUS_COMMAND (778) DMCB1 47
SYSTEM_TASK_PRIORITY 4 SMDCC 356
SYSTEM_TASK_SUSPEND_INTERVAL 4 SMDCC 356
SYSTEM_TASK_SUSPEND_INTERVAL_SOS 4 SMDCC 356
SYSTEM_TASK_SUSPEND_NAME 8 SMDCC 356
SYSTEMLOG (0) L2SL 240
SZAI_ALLOCATE 1 FEP02 114
SZAI_CHAINTO (28) FEP02 114
SZAI_CHAINTO_INVALID 1 FEP02 114
SZAI_CHAINTO_X (11) FEP02 113
SZAI_COLLECT_RESID 1 FEP02 114
SZAI_COLLECT_RESTYPE 1 FEP02 114
SZAI_CONVID (2C) FEP02 114
SZAI_CONVID_INVALID 1 FEP02 114
SZAI_CONVID_X (BIT) FEP02 113
SZAI_DISASTER 1 FEP02 114
SZAI_DISCARD 1 FEP02 114
SZAI_ELEMENT_INVALID 1 FEP02 114
SZAI_ELEMENT_LENGTH 114
SZAI_ELEMENT_LENGTH_X 113
SZAI_EXCEPTION 1 FEP02 114
SZAI_EXISTENCE 113
SZAI_EXTRACT 1 FEP02 114
SZAI_FORMAT_NO 113
SZAI_FQCC (40) FEP02 114
SZAI_FQCC_X (BIT) FEP02 113
SZAI_FREE 1 FEP02 114
SZAI_FREEMAIN_ERROR 1 FEP02 114
SZAI_FUNCTION (18) FEP02 113
SZAI_FUNCTION_X (BIT) FEP02 113
SZAI_GETMAIN_ERROR 1 FEP02 114
SZAI_HEAD (0) FEP02 113
SZAI_INQUIRE 1 FEP02 114
SZAI_INSTALL 1 FEP02 114
SZAI_INVALID 1 FEP02 114
SZAI_ISSUE 1 FEP02 114
SZAI_KERNERROR 1 FEP02 114
SZAI_KERNHANDLE 113
SZAI_LENGTH_INVALID 1 FEP02 114
SZAI_NO_STORAGE 1 FEP02 114
SZAI_NOOP 1 FEP02 114
SZAI_OK 1 FEP02 114
SZAI_PARMLIST_INVALID 1 FEP02 114
SZAI_PLISTLEN (0) FEP02 113
SZAI_PREPARE 1 FEP02 114
SZAI_PURGED 1 FEP02 114
SZAI_QUEUE 1 FEP02 114
SZAI_QUEUE_ELEMENT (24) FEP02 114
SZAI_QUEUE_ELEMENT_X (BIT) FEP02 113
SZAI_REASON (1B) FEP02 113
SZAI_REASON_X (BIT) FEP02 113
SZAI_RECEIVE 1 FEP02 114
SZAI_RELEASE 1 FEP02 114
SZAI_REQUEST 1 FEP02 114
SZAI_REQUEST_INVALID 1 FEP02 114
SZAI_REQUEST_TYPE (1C) FEP02 114
SZAI_REQUEST_TYPE_X (BIT) FEP02 113
SZAI_RESPONSE 113
SZAI_RESPONSE_X 113
SZAI_RM_INACTIVE 1 FEP02 114
SZAI_SEND 1 FEP02 114
SZAI_SET 1 FEP02 114
SZAI_START 1 FEP02 114
SZAI_TASK_NUMBER (3C) FEP02 114
SZAI_TASK_NUMBER_X (BIT) FEP02 113
SZAI_TERMID (34) FEP02 114
SZAI_TERMID_X (BIT) FEP02 113
SZAI_TERMINATE 1 FEP02 114
SZAI_TRANID (38) FEP02 114
SZAI_TRANID_X (BIT) FEP02 113
SZAI_VERSION_NO (8) FEP02 113
SZD_AC_ACB 115
SZD_AC_CPA (24) FEP03 115
SZD_AC_EYE (0) FEP03 115
SZD_AC_NAME (29) FEP03 115
SZD_AC_NAMEL 115
SZD_AC_NEXT (24) FEP03 115
SZD_AC_PASSL (34) FEP03 115
SZD_AC_PASSWORD (35) FEP03 115
SZD_AC_PREV 115
SZD_BI_BINDAREA (30) FEP04 116
SZD_BI_BINDLTH (34) FEP04 116
SZD_BI_CID (2C) FEP04 116
SZD_BI_DELETED (BIT) FEP04 116
SZD_BI_EYE (0) FEP04 116
SZD_BI_FLAGS 116
SZD_BI_I_SEQNO (3C) FEP04 116
SZD_BI_PARMSESS (38) FEP04 116
SZD_BI_PRIMARY_LU_NAME (3E) FEP04 116
SZD_BI_QC (20) FEP04 116
SZD_BI_QCB (20) FEP04 116
SZD_BI_REPORT (BIT) FEP04 116
SZD_BI_WE 116
SZD_CD_ACQSTATUS (EE) FEP05 119
SZD_CD_AGATE (BIT) FEP05 118
SZD_CD_ALLOC (BIT) FEP05 118
SZD_CD_ALLOC_INC (BIT) FEP05 119
SZD_CD_API (C4) FEP05 119
SZD_CD_API_QE (60) FEP05 119
SZD_CD_API_QUEUED (BIT) FEP05 118
SZD_CD_AWAITING_RESPONSE (BIT) FEP05 118
SZD_CD_BID_PURGE (BIT) FEP05 118
SZD_CD_BINDAREA (5C) FEP05 119
SZD_CD_BINDLTH (6C) FEP05 119
SZD_CD_BINDR (BIT) FEP05 118
SZD_CD_BSX_SCHED (BIT) FEP05 118
SZD_CD_CD_SENT (BIT) FEP05 118
SZD_CD_CID (68) FEP05 119
SZD_CD_CLEARR (BIT) FEP05 118
SZD_CD_CLEARREP (BIT) FEP05 118
SZD_CD_CURRENT 119
SZD_CD_CVPTR (E8) FEP05 119
SZD_CD_DATA_DRA (54) FEP05 119
SZD_CD_DATAR (BIT) FEP05 118
SZD_CD_DCNEXT (108) FEP05 119
SZD_CD_DCPREV (104) FEP05 119
SZD_CD_DEL_CONN (BIT) FEP05 119
SZD_CD_DEL_NODE (BIT) FEP05 119
SZD_CD_DEL_POOL (BIT) FEP05 119
SZD_CD_DEL_TARGET (BIT) FEP05 119
SZD_CD_DESSTATUS (F0) FEP05 119
SZD_CD_DEVICE (74) FEP05 119
SZD_CD_DRAINING (BIT) FEP05 118
SZD_CD_DREASON (50) FEP05 119
SZD_CD_DSPTR (100) FEP05 119
SZD_CD_DTR (BIT) FEP05 117
SZD_CD_DYNAM (BIT) FEP05 118
SZD_CD_END (178) FEP05 120
SZD_CD_ERRORS (174) FEP05 120
SZD_CD_EVENTVALUE (70) FEP05 119
SZD_CD_EXREQ (BIT) FEP05 119
SZD_CD_EYE (0) FEP05 117
SZD_CD_FLAGS_ALLOC (44) FEP05 117
SZD_CD_FLAGS_ALLOC1 (44) FEP05 117
SZD_CD_FLAGS_ALLOC2 (45) FEP05 117
SZD_CD_FLAGS_ALLOC3 (46) FEP05 118
SZD_CD_FLAGS_ALLOC4 118
SZD_CD_FLAGS_FP1 (BIT) FEP05 118
SZD_CD_FLAGS_PP1 118
SZD_CD_FLAGS_SC1 118
SZD_CD_FLAGS_SC2 (49) FEP05 118
SZD_CD_FLAGS_SS1 118
SZD_CD_FLAGS_SS2 (4B) FEP05 118
SZD_CD_FLAGS_SS3 (4C) FEP05 118
SZD_CD_FLAGS_TTD1 (4E) FEP05 118
SZD_CD_FLAGS_TTD2 (4F) FEP05 118
SZD_CD_FREE_TRAN (10C) FEP05 119
SZD_CD_FREE_X (BIT) FEP05 118
SZD_CD_FREEF (BIT) FEP05 118
SZD_CD_FREEQD (BIT) FEP05 118
SZD_CD_FREER (BIT) FEP05 118
SZD_CD_FSX_SCHED (BIT) FEP05 118
SZD_CD_GOOD_MORNING (BIT) FEP05 118
SZD_CD_I_SEQNO (7C) FEP05 119
SZD_CD_IBSQAC (7A) FEP05 119
SZD_CD_IBSQVAL (76) FEP05 119
SZD_CD_INB (BIT) FEP05 118
SZD_CD_INSTSTATUS (F2) FEP05 119
SZD_CD_LOFF (BIT) FEP05 117
SZD_CD_LOGMODE (AC) FEP05 119
SZD_CD_LOSE (BIT) FEP05 118
SZD_CD_LOST (BIT) FEP05 117
SZD_CD_LOSTR (BIT) FEP05 118
SZD_CD_MIC (BIT) FEP05 118
SZD_CD_MISC 119
SZD_CD_NDCLOSE (BIT) FEP05 118

SZD_CD_NDNEXT (D0) FEP05   119
SZD_CD_NDPREV (CC) FEP05   119
SZD_CD_NDPTR (E4) FEP05   119
SZD_CD_NEXT (C8) FEP05   119
SZD_CD_NSEXIT_CODE (8C) FEP05   119
SZD_CD_NSEXIT_LTH (84) FEP05   119
SZD_CD_NSEXITR (BIT) FEP05   118
SZD_CD_O_SEQNO (7E) FEP05   119
SZD_CD_OBSQAC (7B) FEP05   119
SZD_CD_OBSQVAL (78) FEP05   119
SZD_CD_ON_REQ (BIT) FEP05   118
SZD_CD_ON_REQIRB (BIT) FEP05   118
SZD_CD_ON_SCQ   117
SZD_CD_ON_SCQIRB (BIT) FEP05   117
SZD_CD_ON_TMR (BIT) FEP05   117
SZD_CD_OPNSEC (BIT) FEP05   118
SZD_CD_OPNSEC_OK (BIT) FEP05   118
SZD_CD_OPNSEC_REJ (BIT) FEP05   118
SZD_CD_PARMSESS (64) FEP05   119
SZD_CD_PDPTR (DC) FEP05   119
SZD_CD_PEND_EB (BIT) FEP05   118
SZD_CD_PEND_MORNING (BIT) FEP05   118
SZD_CD_PENDTR (BIT) FEP05   118
SZD_CD_POS_DRAINING (BIT) FEP05   118
SZD_CD_PREV (C4) FEP05   119
SZD_CD_QC (BIT) FEP05   118
SZD_CD_QEC (BIT) FEP05   117
SZD_CD_RCOUNT (120) FEP05   119
SZD_CD_RCVD_MORNING (BIT) FEP05   118
SZD_CD_RDLEN (11C) FEP05   119
SZD_CD_RDPTR (118) FEP05   119
SZD_CD_RE_QC (38) FEP05   117
SZD_CD_RE_QCB (38) FEP05   117
SZD_CD_RE_REQ   117
SZD_CD_RE_WE   117
SZD_CD_RECEIVED (168) FEP05   119
SZD_CD_RECEIVETIMEOUTS (170) FEP05   120
SZD_CD_RELQ (BIT) FEP05   118
SZD_CD_REQ (BIT) FEP05   118
SZD_CD_REQD (BIT) FEP05   118
SZD_CD_RESP_DRA (58) FEP05   119
SZD_CD_RESPR (BIT) FEP05   118
SZD_CD_RETCODE (80) FEP05   119
SZD_CD_SC_QC (24) FEP05   117
SZD_CD_SC_QCB (20) FEP05   117
SZD_CD_SC_QP (20) FEP05   117
SZD_CD_SC_REQ (28) FEP05   117
SZD_CD_SC_WE   117
SZD_CD_SDT_OK (BIT) FEP05   118
SZD_CD_SDT_REP (BIT) FEP05   118
SZD_CD_SDTR (BIT) FEP05   118
SZD_CD_SDX_SCHED (BIT) FEP05   118
SZD_CD_SENT (164) FEP05   119
SZD_CD_SERVSTATUS (EC) FEP05   119
SZD_CD_SESSSTATUS (F4) FEP05   119
SZD_CD_SHUTC   118
SZD_CD_SHUTD (BIT) FEP05   117
SZD_CD_SIGNON_TRAN (B8) FEP05   119
SZD_CD_SIGNON_X (BIT) FEP05   118
SZD_CD_SIP (BIT) FEP05   118
SZD_CD_SSENSE (114) FEP05   119
SZD_CD_STSN   118
SZD_CD_STSN_OK (BIT) FEP05   118
SZD_CD_STSN_SCHED (BIT) FEP05   118
SZD_CD_STSN_TRAN (BC) FEP05   119
SZD_CD_STSN_X (BIT) FEP05   118
SZD_CD_STSNR (BIT) FEP05   118
SZD_CD_TDNEXT (D8) FEP05   119
SZD_CD_TDPREV (D4) FEP05   119
SZD_CD_TDPTR (E0) FEP05   119
SZD_CD_TDQ (B4) FEP05   119
SZD_CD_TERM_C (BIT) FEP05   117
SZD_CD_TERM_Q   117
SZD_CD_TERM_U (BIT) FEP05   117
SZD_CD_TRINTVL   117
SZD_CD_TRTYPE (32) FEP05   117
SZD_CD_UDATA (124) FEP05   119
SZD_CD_UDFLAG (BIT) FEP05   118
SZD_CD_UDX_SCHED (BIT) FEP05   118
SZD_CD_UNBIND_CODE   119
SZD_CD_UNBIND_LTH (82) FEP05   119
SZD_CD_UNBINDR (BIT) FEP05   118
SZD_CD_UNSOL_TRAN (C0) FEP05   119
SZD_CD_UNSOLD_X (BIT) FEP05   118
SZD_CD_UNSOLICITEDINPUTS (16C) FEP05   120
SZD_CD_URFLAG (BIT) FEP05   118
SZD_CD_USAGE (FC) FEP05   119
SZD_CD_USENSE (110) FEP05   119
SZD_CD_USX_SCHED (BIT) FEP05   118
SZD_CD_XCPTN_X (BIT) FEP05   118
SZD_CM_2DX (BIT) FEP06   122
SZD_CM_2IX (189) FEP06   122
SZD_CM_2OX (BIT) FEP06   122
SZD_CM_2PX (BIT) FEP06   122
SZD_CM_2QX (BIT) FEP06   122
SZD_CM_2SX (BIT) FEP06   122
SZD_CM_ACBTEMP (8C) FEP06   121
SZD_CM_ACTIVE_CVLIST (64) FEP06   121
SZD_CM_BCLIST (9C) FEP06   121
SZD_CM_CDLIST (17C) FEP06   122
SZD_CM_CQE (7C) FEP06   121
SZD_CM_CQECB (11C) FEP06   121
SZD_CM_CQHEAD (140) FEP06   121
SZD_CM_CQPTR (100) FEP06   121
SZD_CM_CQSYS (144) FEP06   121
SZD_CM_CVID   122
SZD_CM_DCQLIST (170) FEP06   121
SZD_CM_DDDLIST (1A0) FEP06   122
SZD_CM_DDLIST (AC) FEP06   121
SZD_CM_DISPK (A8) FEP06   121
SZD_CM_DSTAT (90) FEP06   121
SZD_CM_END (1AC) FEP06   122
SZD_CM_EQECB (114) FEP06   121
SZD_CM_EQHEAD (130) FEP06   121
SZD_CM_EQPTR (F8) FEP06   121
SZD_CM_EQSYS (134) FEP06   121
SZD_CM_EXITMSK (188) FEP06   122
SZD_CM_EXLST (88) FEP06   121
SZD_CM_EYE (0) FEP06   120
SZD_CM_FLAGS (92) FEP06   121
SZD_CM_FREE_QCB (48) FEP06   120
SZD_CM_FREE_QUEUE (48) FEP06   120
SZD_CM_INACTIVE_CVLIST (68) FEP06   121
SZD_CM_IQECB (120) FEP06   121
SZD_CM_IQHEAD (148) FEP06   121
SZD_CM_IQPTR (104) FEP06   121
SZD_CM_IQSYS (14C) FEP06   121
SZD_CM_IRBLEN (180) FEP06   122
SZD_CM_IRBSAVE (4C) FEP06   120
SZD_CM_LIFO (60) FEP06   121
SZD_CM_LIFOLEN (184) FEP06   122
SZD_CM_NDLIST (6C) FEP06   121
SZD_CM_NIB_MASK (5C) FEP06   121
SZD_CM_OPNSEC_MASK (54) FEP06   121
SZD_CM_PDLIST (74) FEP06   121
SZD_CM_PDX (BIT) FEP06   122
SZD_CM_PIX (18A) FEP06   122
SZD_CM_POX (BIT) FEP06   122
SZD_CM_PQX (BIT) FEP06   122
SZD_CM_PSLIST (78) FEP06   121
SZD_CM_PSX (BIT) FEP06   122
SZD_CM_QECBLIST (F8) FEP06   121
SZD_CM_RASIZE (98) FEP06   121
SZD_CM_RECANY_MASK (58) FEP06   121
SZD_CM_RETRY (174) FEP06   121
SZD_CM_RETRY1 (194) FEP06   122
SZD_CM_RETRY2 (198) FEP06   122
SZD_CM_RETRYK (176) FEP06   121
SZD_CM_RLIM (19C) FEP06   122
SZD_CM_RMID (190) FEP06   122
SZD_CM_RPL_MASK (50) FEP06   120
SZD_CM_SC_ECBIRB (124) FEP06   121
SZD_CM_SC_ECBIRBT (128) FEP06   121
SZD_CM_SC_ECBTPEND8 (12C) FEP06   121
SZD_CM_SC_PTRIRB (108) FEP06   121
SZD_CM_SC_PTRIRBT (10C) FEP06   121
SZD_CM_SC_PTRTPEND8 (110) FEP06   121
SZD_CM_SC_QC (20) FEP06   120
SZD_CM_SC_QCB   120
SZD_CM_SC_QCBIRB (38) FEP06   120
SZD_CM_SC_QCBIRBT (30) FEP06   120
SZD_CM_SC_QCBT (28) FEP06   120
SZD_CM_SC_QCBTPEND8 (40) FEP06   120
SZD_CM_SC_QCIRB (38) FEP06   120
SZD_CM_SC_QCIRBT (30) FEP06   120
SZD_CM_SC_QCT (28) FEP06   120

SZD_CM_SC_QCTPEND8 (40) FEP06 120
SZD_CM_SC_SYS (24) FEP06 120
SZD_CM_SC_SYSIRB (3C) FEP06 120
SZD_CM_SC_SYSIRBT (34) FEP06 120
SZD_CM_SC_SYST (2C) FEP06 120
SZD_CM_SC_SYSTPEND8 (44) FEP06 120
SZD_CM_SCHEDPPM (BIT) FEP06 121
SZD_CM_SCHEDTQA (BIT) FEP06 121
SZD_CM_SDS (84) FEP06 121
SZD_CM_STECB (1A8) FEP06 122
SZD_CM_STEXIT (15C) FEP06 121
SZD_CM_STFLAGS (150) FEP06 121
SZD_CM_STIMERM_ECB (1A4) FEP06 122
SZD_CM_STIMERM_PARMS (150) FEP06 121
SZD_CM_STIMFAIL (BIT) FEP06 121
SZD_CM_STPARM (160) FEP06 121
SZD_CM_STPTR (1A4) FEP06 122
SZD_CM_TDLIST (70) FEP06 121
SZD_CM_TICK (A4) FEP06 121
SZD_CM_TICKID (16C) FEP06 121
SZD_CM_TICKIDA (154) FEP06 121
SZD_CM_TICKLEN 121
SZD_CM_TICKPTR (158) FEP06 121
SZD_CM_TOLIST (A0) FEP06 121
SZD_CM_TQALIST (178) FEP06 122
SZD_CM_TQE (80) FEP06 121
SZD_CM_WAITK (94) FEP06 121
SZD_CM_WSL (BIT) FEP06 122
SZD_CM_XDA (BIT) FEP06 122
SZD_CM_XFR (BIT) FEP06 122
SZD_CM_XLT (BIT) FEP06 122
SZD_CM_XNS (BIT) FEP06 122
SZD_CM_XQECB (118) FEP06 121
SZD_CM_XQHEAD (138) FEP06 121
SZD_CM_XQPTR (FC) FEP06 121
SZD_CM_XQSYS (13C) FEP06 121
SZD_CM_XRA (BIT) FEP06 122
SZD_CM_XSC (BIT) FEP06 122
SZD_CM_XTP (BIT) FEP06 122
SZD_CM_YQR 122
SZD_CM_YRI (BIT) FEP06 122
SZD_CM_YSC (BIT) FEP06 122
SZD_CM_YSR (BIT) FEP06 122
SZD_CM_YSY (BIT) FEP06 122
SZD_CV_APIQ (44) FEP07 126
SZD_CV_BROWSE (BIT) FEP07 126
SZD_CV_BSIZE (2C) FEP07 125
SZD_CV_BTPTR (70) FEP07 126
SZD_CV_BTSIZE 126
SZD_CV_CDPTR (28) FEP07 125
SZD_CV_ECOUNT (6C) FEP07 126
SZD_CV_EYE (0) FEP07 125
SZD_CV_FLAGS 126
SZD_CV_FQCC (4C) FEP07 126
SZD_CV_ID (30) FEP07 126
SZD_CV_IDX (30) FEP07 126
SZD_CV_IDY (34) FEP07 126
SZD_CV_NDPTR (30) FEP07 126
SZD_CV_NEXT (24) FEP07 125
SZD_CV_PDPTR (2C) FEP07 125
SZD_CV_PREV 125
SZD_CV_PSPTR (2C) FEP07 126
SZD_CV_RTYPE (44) FEP07 126
SZD_CV_TASK_NUM (40) FEP07 126
SZD_CV_TDPTR (34) FEP07 126
SZD_CV_TERMID (3C) FEP07 126
SZD_CV_TID (38) FEP07 126
SZD_CV_TRANID (38) FEP07 126
SZD_DS_AFLAG (BIT) FEP08 128
SZD_DS_AID (92) FEP08 129
SZD_DS_ALARM (BIT) FEP08 128
SZD_DS_ATLIM (9C) FEP08 129
SZD_DS_BFLAG (A0) FEP08 129
SZD_DS_BG (BIT) FEP08 130
SZD_DS_CBA (4C) FEP08 127
SZD_DS_CBG (BIT) FEP08 129
SZD_DS_CC (90) FEP08 128
SZD_DS_CCBYTE (94) FEP08 129
SZD_DS_CCP (48) FEP08 127
SZD_DS_CDPTR (5C) FEP08 127
SZD_DS_CFG (BIT) FEP08 129
SZD_DS_CFO (BIT) FEP08 129
SZD_DS_CFV (BIT) FEP08 129
SZD_DS_CHAIN (7C) FEP08 127
SZD_DS_CMD (BIT) FEP08 128
SZD_DS_COLOUR (BIT) FEP08 129
SZD_DS_CONTROL (8C) FEP08 128
SZD_DS_CPPROT (BIT) FEP08 128
SZD_DS_CSBYTE (96) FEP08 129
SZD_DS_CVBYTE (97) FEP08 129
SZD_DS_CXA (BIT) FEP08 129
SZD_DS_CXBYTE (95) FEP08 129
SZD_DS_CXP (BIT) FEP08 129
SZD_DS_DABYTE (9E) FEP08 129
SZD_DS_DBA (54) FEP08 127
SZD_DS_DBG (BIT) FEP08 129
SZD_DS_DCBYTE (98) FEP08 129
SZD_DS_DFG (BIT) FEP08 129
SZD_DS_DFLAGS (EC) FEP08 129
SZD_DS_DFLEN (BIT) FEP08 129
SZD_DS_DFO (BIT) FEP08 129
SZD_DS_DFV (BIT) FEP08 129
SZD_DS_DLENGTH (60) FEP08 127
SZD_DS_DS1 (BIT) FEP08 130
SZD_DS_DS2 (BIT) FEP08 130
SZD_DS_DSBYTE (9A) FEP08 129
SZD_DS_DVBYTE (9B) FEP08 129
SZD_DS_DXA (BIT) FEP08 129
SZD_DS_DXBYTE (99) FEP08 129
SZD_DS_DXP (BIT) FEP08 129
SZD_DS_EDS (BIT) FEP08 129
SZD_DS_END (F4) FEP08 129
SZD_DS_ERI (BIT) FEP08 129
SZD_DS_EU (BIT) FEP08 128
SZD_DS_EU1 (BIT) FEP08 128
SZD_DS_EYE (0) FEP08 127
SZD_DS_FG (BIT) FEP08 130
SZD_DS_FLAG3 (8D) FEP08 128
SZD_DS_FLAGS (2C) FEP08 127
SZD_DS_FO (BIT) FEP08 130
SZD_DS_FV (BIT) FEP08 130
SZD_DS_GATE (BIT) FEP08 128
SZD_DS_GE (BIT) FEP08 128
SZD_DS_IDATA (74) FEP08 127
SZD_DS_IDLEN (78) FEP08 127
SZD_DS_IDPTR (6C) FEP08 127
SZD_DS_IFLAG (BIT) FEP08 128
SZD_DS_INOP (BIT) FEP08 128
SZD_DS_INPID (93) FEP08 129
SZD_DS_INS (BIT) FEP08 128
SZD_DS_KINDEX (64) FEP08 127
SZD_DS_KLOCK (BIT) FEP08 128
SZD_DS_L1PROT (BIT) FEP08 128
SZD_DS_LA (68) FEP08 127
SZD_DS_MDPTR (70) FEP08 127
SZD_DS_MDR (BIT) FEP08 128
SZD_DS_MDT 130
SZD_DS_MF (BIT) FEP08 128
SZD_DS_MSIP (BIT) FEP08 129
SZD_DS_NEXT (24) FEP08 127
SZD_DS_NFIP (BIT) FEP08 129
SZD_DS_NUM (BIT) FEP08 130
SZD_DS_P1APTR (34) FEP08 127
SZD_DS_P1CPTR (44) FEP08 127
SZD_DS_P1GPTR (30) FEP08 127
SZD_DS_P1SPTR (3C) FEP08 127
SZD_DS_P1VPTR (40) FEP08 127
SZD_DS_P1XPTR (38) FEP08 127
SZD_DS_PBB (BIT) FEP08 128
SZD_DS_PFLIM (9D) FEP08 129
SZD_DS_POST (BIT) FEP08 128
SZD_DS_PREV 127
SZD_DS_PROT (BIT) FEP08 130
SZD_DS_PSI (BIT) FEP08 128
SZD_DS_PSIZE (80) FEP08 128
SZD_DS_PSX (84) FEP08 128
SZD_DS_PSXALT (88) FEP08 128
SZD_DS_PSXDEF (86) FEP08 128
SZD_DS_PSY (85) FEP08 128
SZD_DS_PSYALT (89) FEP08 128
SZD_DS_PSYDEF (87) FEP08 128
SZD_DS_QCODE (AB) FEP08 129
SZD_DS_QDATA (AC) FEP08 129
SZD_DS_QID (AA) FEP08 129
SZD_DS_QLEN (A8) FEP08 129
SZD_DS_QP_ALPHA (BIT) FEP08 129

SZD_DS_QP_ASIA (BIT) FEP08 129
SZD_DS_QP_CHARS (BIT) FEP08 129
SZD_DS_QP_COLOR (BIT) FEP08 129
SZD_DS_QP_FLAG1 (ED) FEP08 129
SZD_DS_QP_FLAG2 (EE) FEP08 129
SZD_DS_QP_HILI (BIT) FEP08 129
SZD_DS_QP_IMPA (BIT) FEP08 129
SZD_DS_QP_OUTL (BIT) FEP08 129
SZD_DS_QP_SUMM (BIT) FEP08 129
SZD_DS_QP_TRAN (BIT) FEP08 129
SZD_DS_QP_USEA (BIT) FEP08 129
SZD_DS_QP_VALI (BIT) FEP08 129
SZD_DS_RA (BIT) FEP08 128
SZD_DS_RA1 (BIT) FEP08 128
SZD_DS_RA2 (BIT) FEP08 128
SZD_DS_RDPTR (F0) FEP08 129
SZD_DS_RIP (BIT) FEP08 129
SZD_DS_RMT (BIT) FEP08 128
SZD_DS_SA (BIT) FEP08 128
SZD_DS_SAT (A1) FEP08 129
SZD_DS_SB (BIT) FEP08 128
SZD_DS_SB1 (BIT) FEP08 128
SZD_DS_SE (BIT) FEP08 128
SZD_DS_SEC (AE) FEP08 129
SZD_DS_SENDREQ (BIT) FEP08 128
SZD_DS_SENSE (58) FEP08 127
SZD_DS_SEQ1 (8E) FEP08 128
SZD_DS_SEQ2 (8F) FEP08 128
SZD_DS_SET (AF) FEP08 129
SZD_DS_SF (BIT) FEP08 128
SZD_DS_SFDATA (A8) FEP08 129
SZD_DS_SFID (A4) FEP08 129
SZD_DS_SFID2 (A5) FEP08 129
SZD_DS_SFL1 (BIT) FEP08 129
SZD_DS_SFL2 (BIT) FEP08 129
SZD_DS_SFLEN (A2) FEP08 129
SZD_DS_SFLEN1 (A2) FEP08 129
SZD_DS_SFLEN2 (A3) FEP08 129
SZD_DS_SFPID (A6) FEP08 129
SZD_DS_SFPIDX 129
SZD_DS_SFTYPE (A7) FEP08 129
SZD_DS_SLOCK (BIT) FEP08 128
SZD_DS_TB1 (AC) FEP08 129
SZD_DS_TB2 (AD) FEP08 129
SZD_DS_TBA (50) FEP08 127
SZD_DS_TPS (BIT) FEP08 129
SZD_DS_TWAIT (BIT) FEP08 128
SZD_DS_TYPE (28) FEP08 127
SZD_DS_WC (91) FEP08 128
SZD_DS_WC_ALARM (BIT) FEP08 128
SZD_DS_WC_KENA (BIT) FEP08 129
SZD_DS_WC_P1 (BIT) FEP08 128
SZD_DS_WC_P2 (BIT) FEP08 128
SZD_DS_WC_RESET 128
SZD_DS_WC_RMDT (BIT) FEP08 129
SZD_DS_WC_SP (BIT) FEP08 128
SZD_DS_WSFCC (9F) FEP08 129
SZD_DS_WSFIP (BIT) FEP08 128
SZD_DS_WSFREQ (BIT) FEP08 128
SZD_DS_XA (BIT) FEP08 130
SZD_DS_XP (BIT) FEP08 130
SZD_EC_CBID (18) FEP03 115
SZD_EC_CBID (18) FEP04 116
SZD_EC_CBID (18) FEP05 117
SZD_EC_CBID (18) FEP06 120, 122, 123, 124
SZD_EC_CBID (18) FEP07 125
SZD_EC_CBID (18) FEP08 127
SZD_EC_CBID (18) FEP09 131
SZD_EC_CBID (18) FEP10 132
SZD_EC_CBID (18) FEP11 134
SZD_EC_CBID (18) FEP12 135
SZD_EC_CBID (18) FEP13 136
SZD_EC_CBID (18) FEP14 138
SZD_EC_CBID (18) FEP15 139
SZD_EC_CBID (18) FEP16 140
SZD_EC_CBID (18) FEP17 141
SZD_EC_CBID (18) FEP18 145
SZD_EC_CBID (18) FEP19 146
SZD_EC_CBID (18) FEP20 147
SZD_EC_GT (2) FEP03 115
SZD_EC_GT (2) FEP04 116
SZD_EC_GT (2) FEP05 117
SZD_EC_GT (2) FEP06 120, 122, 123, 124
SZD_EC_GT (2) FEP07 125
SZD_EC_GT (2) FEP08 127
SZD_EC_GT (2) FEP09 131
SZD_EC_GT (2) FEP10 132
SZD_EC_GT (2) FEP11 134
SZD_EC_GT (2) FEP12 135
SZD_EC_GT (2) FEP13 136
SZD_EC_GT (2) FEP14 138
SZD_EC_GT (2) FEP15 139
SZD_EC_GT (2) FEP16 140
SZD_EC_GT (2) FEP17 141
SZD_EC_GT (2) FEP18 145
SZD_EC_GT (2) FEP19 146
SZD_EC_GT (2) FEP20 147
SZD_EC_LENGTH (0) FEP03 115
SZD_EC_LENGTH (0) FEP04 116
SZD_EC_LENGTH (0) FEP05 117
SZD_EC_LENGTH (0) FEP06 120, 122, 123, 124
SZD_EC_LENGTH (0) FEP07 125
SZD_EC_LENGTH (0) FEP08 127
SZD_EC_LENGTH (0) FEP09 131
SZD_EC_LENGTH (0) FEP10 132
SZD_EC_LENGTH (0) FEP11 134
SZD_EC_LENGTH (0) FEP12 135
SZD_EC_LENGTH (0) FEP13 136
SZD_EC_LENGTH (0) FEP14 138
SZD_EC_LENGTH (0) FEP15 139
SZD_EC_LENGTH (0) FEP16 140
SZD_EC_LENGTH (0) FEP17 141
SZD_EC_LENGTH (0) FEP18 145
SZD_EC_LENGTH (0) FEP19 146
SZD_EC_LENGTH (0) FEP20 147
SZD_EC_NAME (3) FEP03 115
SZD_EC_NAME (3) FEP04 116
SZD_EC_NAME (3) FEP05 117
SZD_EC_NAME (3) FEP06 120, 122, 123, 124
SZD_EC_NAME (3) FEP07 125
SZD_EC_NAME (3) FEP08 127
SZD_EC_NAME (3) FEP09 131
SZD_EC_NAME (3) FEP10 132
SZD_EC_NAME (3) FEP11 134
SZD_EC_NAME (3) FEP12 135
SZD_EC_NAME (3) FEP13 136
SZD_EC_NAME (3) FEP14 138
SZD_EC_NAME (3) FEP15 139
SZD_EC_NAME (3) FEP16 140
SZD_EC_NAME (3) FEP17 141
SZD_EC_NAME (3) FEP18 145
SZD_EC_NAME (3) FEP19 146
SZD_EC_NAME (3) FEP20 147
SZD_EC_SPID 115, 116, 117, 120, 122, 123, 124, 125, 127, 131, 132, 134, 135, 136, 138, 139, 140, 141, 145, 146, 147
SZD_IDQ_EYE (0) FEP06 124
SZD_IDQ_QNEXT 124
SZD_IDQ_QREQ (0) FEP06 123
SZD_KESTACK_SAVE (B0) FEP06 121
SZD_ND_ACB (58) FEP10 133
SZD_ND_ACPTR (60) FEP10 133
SZD_ND_ACQSTATUS (7E) FEP10 133
SZD_ND_API (48) FEP10 133
SZD_ND_ASTAT (84) FEP10 133
SZD_ND_BI_QC (38) FEP10 132
SZD_ND_BI_QCB (38) FEP10 132
SZD_ND_CDLIST (50) FEP10 133
SZD_ND_CDSTQ (88) FEP10 133
SZD_ND_CLOSE (BIT) FEP10 132
SZD_ND_CM (5C) FEP10 133
SZD_ND_DEFTRAN (34) FEP10 132
SZD_ND_DESSTATUS (80) FEP10 133
SZD_ND_DISCARD (BIT) FEP10 132
SZD_ND_ERFLG (86) FEP10 133
SZD_ND_EYE (0) FEP10 132
SZD_ND_FLAGS 132
SZD_ND_IMMED (BIT) FEP10 132
SZD_ND_INSTSTATUS (82) FEP10 133
SZD_ND_NAME (65) FEP10 133
SZD_ND_NAMEL 133
SZD_ND_NEXT (4C) FEP10 133
SZD_ND_ON_Q 132
SZD_ND_ON_QIRB (BIT) FEP10 132
SZD_ND_ON_QTPEND8 132
SZD_ND_ON_TMR (BIT) FEP10 132
SZD_ND_OPENFAIL (BIT) FEP10 132

SZD_ND_OPENOK (BIT) FEP10 132
SZD_ND_OPENREQ (BIT) FEP10 132
SZD_ND_OPENRIP (BIT) FEP10 132
SZD_ND_PASSL (70) FEP10 133
SZD_ND_PASSWORD (71) FEP10 133
SZD_ND_PREV (48) FEP10 133
SZD_ND_QC (24) FEP10 132
SZD_ND_QCB (20) FEP10 132
SZD_ND_QP (20) FEP10 132
SZD_ND_RADONE (BIT) FEP10 133
SZD_ND_RCOUNT (90) FEP10 133
SZD_ND_RECANY (44) FEP10 133
SZD_ND_RECANYN (BIT) FEP10 132
SZD_ND_RECANYR (BIT) FEP10 132
SZD_ND_REQ (28) FEP10 132
SZD_ND_SERVSTATUS 133
SZD_ND_SHUT (41) FEP10 132
SZD_ND_SLDONE (43) FEP10 133
SZD_ND_SLFAIL (BIT) FEP10 132
SZD_ND_SLMEM (BIT) FEP10 132
SZD_ND_SRLIST (54) FEP10 133
SZD_ND_TPEND (BIT) FEP10 132
SZD_ND_TPEND_0 (BIT) FEP10 132
SZD_ND_TPEND_4 (BIT) FEP10 132
SZD_ND_TPEND_8 (BIT) FEP10 132
SZD_ND_TRINTVL (30) FEP10 132
SZD_ND_TRTYPE (32) FEP10 132
SZD_ND_UDATA (94) FEP10 133
SZD_ND_UNSOL (42) FEP10 133
SZD_ND_UNSOLEX (BIT) FEP10 133
SZD_ND_USAGE (8C) FEP10 133
SZD_ND_WE 132
SZD_PD_ALLOCATED (120) FEP11 135
SZD_PD_ALLOCATESWAITING (12C) FEP11 135
SZD_PD_AWLIST (44) FEP11 134
SZD_PD_CDLIST (40) FEP11 134
SZD_PD_CONNECTIONS (118) FEP11 135
SZD_PD_EYE (0) FEP11 134
SZD_PD_INSTSTATUS (4A) FEP11 134
SZD_PD_NAME (28) FEP11 134
SZD_PD_NDLIST (38) FEP11 134
SZD_PD_NEXT (24) FEP11 134
SZD_PD_NODES (114) FEP11 135
SZD_PD_PKALLOCATED (124) FEP11 135
SZD_PD_PKALLOCATESWAITING (130) FEP11 135
SZD_PD_PKCONNECTIONS (11C) FEP11 135
SZD_PD_PREV 134
SZD_PD_PROPERTY (30) FEP11 134
SZD_PD_PROPS (4C) FEP11 134
SZD_PD_SERVSTATUS (48) FEP11 134
SZD_PD_TARGETS (110) FEP11 135
SZD_PD_TDLIST (3C) FEP11 134
SZD_PD_TIMEOUTS (138) FEP11 135
SZD_PD_TOTALLOCATES (128) FEP11 135
SZD_PD_TOTALLOCATEWAITS (134) FEP11 135
SZD_PD_UDATA (D0) FEP11 135
SZD_PP_BEGINSESSION (48) FEP12 136
SZD_PP_BEGINSESSION_X (BIT) FEP12 135
SZD_PP_CONTENTION (28) FEP12 136
SZD_PP_DEVICE 136
SZD_PP_ENDSESSION (54) FEP12 136
SZD_PP_ENDSESSION_X 135
SZD_PP_EXCEPTIONQ (50) FEP12 136
SZD_PP_EXCEPTIONQ_X 135
SZD_PP_EYE (0) FEP12 135
SZD_PP_FJOURNALNAME (60) FEP12 136
SZD_PP_FJOURNALNUM 136
SZD_PP_FLAGS 135
SZD_PP_FORMAT (26) FEP12 136
SZD_PP_INITIALDATA (2A) FEP12 136
SZD_PP_MAXFLENGTH 136
SZD_PP_MSGJRNL (2C) FEP12 136
SZD_PP_STSN (44) FEP12 136
SZD_PP_STSN_X (BIT) FEP12 135
SZD_PP_UNSOLDATA (4C) FEP12 136
SZD_PP_UNSOLDATA_X (BIT) FEP12 135
SZD_PP_UNSOLDATACK (2E) FEP12 136
SZD_PS_BEGINSESSION (60) FEP13 137
SZD_PS_BEGINSESSION (7C) FEP11 134
SZD_PS_BEGINSESSION_X (BIT) FEP11 134
SZD_PS_BEGINSESSION_X (BIT) FEP13 137
SZD_PS_CONTENTION (54) FEP13 137
SZD_PS_CONTENTION (70) FEP11 134
SZD_PS_DEFTRAN (40) FEP13 137
SZD_PS_DEFTRAN (5C) FEP11 134
SZD_PS_DEVICE (50) FEP13 137
SZD_PS_DEVICE (6C) FEP11 134
SZD_PS_ENDSESSION 134, 137
SZD_PS_ENDSESSION_X 134, 137
SZD_PS_EXCEPTIONQ (68) FEP13 137
SZD_PS_EXCEPTIONQ (84) FEP11 135
SZD_PS_EXCEPTIONQ_X 134, 137
SZD_PS_EYE (0) FEP13 136
SZD_PS_FJOURNALNAME (48) FEP13 137
SZD_PS_FJOURNALNAME (64) FEP11 134
SZD_PS_FLAGS (30) FEP13 136
SZD_PS_FLAGS (4C) FEP11 134
SZD_PS_FORMAT (52) FEP13 137
SZD_PS_FORMAT (6E) FEP11 134
SZD_PS_INITIALDATA (56) FEP13 137
SZD_PS_INITIALDATA (72) FEP11 134
SZD_PS_MAXFLENGTH (44) FEP13 137
SZD_PS_MAXFLENGTH (60) FEP11 134
SZD_PS_MSGJRNL (5A) FEP13 137
SZD_PS_MSGJRNL (76) FEP11 134
SZD_PS_NAME (28) FEP13 136
SZD_PS_NEXT (24) FEP13 136
SZD_PS_PREV 136
SZD_PS_PROPS (30) FEP13 136
SZD_PS_STSN (5C) FEP13 137
SZD_PS_STSN (78) FEP11 134
SZD_PS_STSN_X (BIT) FEP11 134
SZD_PS_STSN_X (BIT) FEP13 137
SZD_PS_UDATA 135, 137
SZD_PS_UNSOLDATA (64) FEP13 137
SZD_PS_UNSOLDATA (80) FEP11 135
SZD_PS_UNSOLDATA_X (BIT) FEP11 134
SZD_PS_UNSOLDATA_X (BIT) FEP13 137
SZD_PS_UNSOLDATACK (58) FEP13 137
SZD_PS_UNSOLDATACK (74) FEP11 134
SZD_QE_CHAIN (34) FEP14 138
SZD_QE_CONFDATA (BIT) FEP14 138
SZD_QE_CONVID (38) FEP14 138
SZD_QE_CVPTR (74) FEP14 138
SZD_QE_DATA (6C) FEP14 138
SZD_QE_DATALEN (70) FEP14 138
SZD_QE_ECB (40) FEP14 138
SZD_QE_EXPFLAG (BIT) FEP14 138
SZD_QE_EYE (0) FEP14 138
SZD_QE_FQCC (44) FEP14 138
SZD_QE_NEXT (24) FEP14 138
SZD_QE_ON_API (BIT) FEP14 138
SZD_QE_ON_IRB (BIT) FEP14 138
SZD_QE_ON_PRB (BIT) FEP14 138
SZD_QE_ON_TMR (BIT) FEP14 138
SZD_QE_ON_TP8 (BIT) FEP14 138
SZD_QE_POSTED (BIT) FEP14 138
SZD_QE_PREFIX (0) FEP14 138
SZD_QE_PREV 138
SZD_QE_PRIVATE (6C) FEP14 138
SZD_QE_PUBLIC (28) FEP14 138
SZD_QE_PURGE (BIT) FEP14 138
SZD_QE_REQDATA (30) FEP14 138
SZD_QE_REQFLAG (2C) FEP14 138
SZD_QE_REQFLAG_POST (BIT) FEP14 138
SZD_QE_REQTYPE (28) FEP14 138
SZD_QE_RP (8C) FEP14 139
SZD_QE_RRT_SEEN (BIT) FEP14 138
SZD_QE_TARGET (88) FEP14 139
SZD_QE_TASKNUM (68) FEP14 138
SZD_QE_TERMID (64) FEP14 138
SZD_QE_TICK (7C) FEP14 139
SZD_QE_TID 138
SZD_QE_TIMED 138
SZD_QE_TIMED_OUT (BIT) FEP14 138
SZD_QE_TNEXT (84) FEP14 139
SZD_QE_TOCK (78) FEP14 139
SZD_QE_TPREV (80) FEP14 139
SZD_QE_TRANID (60) FEP14 138
SZD_RA_CD (3C) FEP15 139
SZD_RA_CM (38) FEP15 139
SZD_RA_DYNAA (34) FEP15 139
SZD_RA_DYNAL (44) FEP15 139
SZD_RA_EYE (0) FEP15 139
SZD_RA_FLAGS (2C) FEP15 139
SZD_RA_ND (40) FEP15 139

SZD_RA_QEB 139
SZD_RA_QNEXT 139
SZD_RA_REQTYPE 139
SZD_RA_RPL (48) FEP15 139
SZD_RA_TRINTVL (30) FEP15 139
SZD_RA_TRTYPE (32) FEP15 139
SZD_RA_VTAM (48) FEP15 139
SZD_RB_CD (3C) FEP16 140
SZD_RB_CM (38) FEP16 140
SZD_RB_DYNAA (34) FEP16 140
SZD_RB_DYNAL (44) FEP16 140
SZD_RB_EYE (0) FEP16 140
SZD_RB_FLAGS (2C) FEP16 140
SZD_RB_ND (40) FEP16 140
SZD_RB_QEB 140
SZD_RB_QNEXT 140
SZD_RB_REQTYPE 140
SZD_RB_RPL (48) FEP16 140
SZD_RB_TRINTVL (30) FEP16 140
SZD_RB_TRTYPE (32) FEP16 140
SZD_RB_VTAM (48) FEP16 140
SZD_REGS_SAVE (B8) FEP06 121
SZD_RIA 141
SZD_RIA_ACQSTATUS (32) FEP17 142
SZD_RIA_AID (31) FEP17 142
SZD_RIA_APPLLIST (48) FEP17 142
SZD_RIA_BEND (BIT) FEP17 141
SZD_RIA_BNEXT (BIT) FEP17 141
SZD_RIA_BNEXTNODE (BIT) FEP17 141
SZD_RIA_BNEXTTARGET (BIT) FEP17 141
SZD_RIA_BSTART (BIT) FEP17 141
SZD_RIA_CHAIN (BIT) FEP17 141
SZD_RIA_COLLECT (31) FEP17 142
SZD_RIA_CONTROL (30) FEP17 141
SZD_RIA_CONVERSE (BIT) FEP17 141
SZD_RIA_CONVID (50) FEP17 142
SZD_RIA_CURSOR (48) FEP17 142
SZD_RIA_CURSOR_X (BIT) FEP17 141
SZD_RIA_DATA (40) FEP17 142
SZD_RIA_DATALEN (3C) FEP17 142
SZD_RIA_ENDTASK (BIT) FEP17 141
SZD_RIA_EOD 142
SZD_RIA_ESCAPE (31) FEP17 142
SZD_RIA_FIELDLOC (44) FEP17 142
SZD_RIA_FIELDNUM (44) FEP17 142
SZD_RIA_FLGS 141
SZD_RIA_FMH (BIT) FEP17 141
SZD_RIA_FORCE (BIT) FEP17 141
SZD_RIA_IMMEDIATE (BIT) FEP17 141
SZD_RIA_INC1 (50) FEP17 142
SZD_RIA_INC2 (58) FEP17 142
SZD_RIA_INC3 (60) FEP17 142
SZD_RIA_INVITE (BIT) FEP17 141
SZD_RIA_KEYSTROKES (BIT) FEP17 141
SZD_RIA_LOCATION (BIT) FEP17 141
SZD_RIA_LST3 (40) FEP17 142
SZD_RIA_LST4 (44) FEP17 142
SZD_RIA_LST5 (48) FEP17 142
SZD_RIA_MAXFLENGTH (3C) FEP17 142
SZD_RIA_NODE (60) FEP17 142
SZD_RIA_NODELIST (44) FEP17 142
SZD_RIA_NODENUM (3C) FEP17 142
SZD_RIA_OPT1 (30) FEP17 141
SZD_RIA_OPT2 (32) FEP17 142
SZD_RIA_PASS (BIT) FEP17 141
SZD_RIA_PASSCONVID (50) FEP17 142
SZD_RIA_PASSWORDLIST (48) FEP17 142
SZD_RIA_POOL (50) FEP17 142
SZD_RIA_POOLLIST (40) FEP17 142
SZD_RIA_POOLNUM (38) FEP17 142
SZD_RIA_PROPERTYSET (60) FEP17 142
SZD_RIA_PROPS (48) FEP17 142
SZD_RIA_RELEASE (BIT) FEP17 141
SZD_RIA_REQSUB (20) FEP17 141
SZD_RIA_REQSUB_CONN 2 FEP17 144
SZD_RIA_REQSUB_CONV 2 FEP17 144
SZD_RIA_REQSUB_CTRL 2 FEP17 144
SZD_RIA_REQSUB_DATA 2 FEP17 144
SZD_RIA_REQSUB_FLD 2 FEP17 144
SZD_RIA_REQSUB_FMT 2 FEP17 144
SZD_RIA_REQSUB_NODE 2 FEP17 144
SZD_RIA_REQSUB_NULL 2 FEP17 144
SZD_RIA_REQSUB_PCHG 2 FEP17 144
SZD_RIA_REQSUB_POOL 2 FEP17 144
SZD_RIA_REQSUB_PROP 2 FEP17 144
SZD_RIA_REQSUB_STSN 2 FEP17 144
SZD_RIA_REQSUB_TGT 2 FEP17 144
SZD_RIA_REQTYPE 141
SZD_RIA_RESET (30) FEP17 141
SZD_RIA_RU (BIT) FEP17 141
SZD_RIA_SENSEDATA (38) FEP17 142
SZD_RIA_SERVSTATUS (30) FEP17 141
SZD_RIA_STATS (40) FEP17 142
SZD_RIA_TARGET (58) FEP17 142
SZD_RIA_TARGETLIST (40) FEP17 142
SZD_RIA_TARGETNUM (38) FEP17 142
SZD_RIA_TERMID (64) FEP17 142
SZD_RIA_TIMEOUT (48) FEP17 142
SZD_RIA_TRANSID (60) FEP17 142
SZD_RIA_USERDATA (4C) FEP17 142
SZD_RIA_VAL1 142
SZD_RIA_VAL2 (3C) FEP17 142
SZD_RIA_VAL3 (40) FEP17 142
SZD_RIA_VAL4 (44) FEP17 142
SZD_RIA_VAL5 (48) FEP17 142
SZD_RIA_VAL6 (4C) FEP17 142
SZD_RIA_VALUE (32) FEP17 142
SZD_ROA 142
SZD_ROA_ACQNUM (8C) FEP17 143
SZD_ROA_ACQSTATUS (7A) FEP17 143
SZD_ROA_ALARMSTATUS (7C) FEP17 143
SZD_ROA_APPL (A0) FEP17 143
SZD_ROA_ATTRS (B0) FEP17 143
SZD_ROA_BACKGROUND (B6) FEP17 144
SZD_ROA_COLOR (B0) FEP17 143
SZD_ROA_COLUMNS (9C) FEP17 143
SZD_ROA_CONVID (A0) FEP17 143
SZD_ROA_CONVNUM (90) FEP17 143
SZD_ROA_CURSOR (94) FEP17 143
SZD_ROA_DATALEN (90) FEP17 143
SZD_ROA_DEVICE (80) FEP17 143
SZD_ROA_ENDSTATUS (78) FEP17 142
SZD_ROA_ESMREASON (90) FEP17 143
SZD_ROA_ESMRESP (8C) FEP17 143
SZD_ROA_FDBK1 (70) FEP17 142
SZD_ROA_FDBK2 (74) FEP17 142
SZD_ROA_FIELDATTR (B7) FEP17 144
SZD_ROA_FIELDS (8C) FEP17 143
SZD_ROA_FMHSTATUS (7C) FEP17 143
SZD_ROA_FORMAT (88) FEP17 143
SZD_ROA_HILIGHT (B1) FEP17 143
SZD_ROA_INPUTCONTROL 143
SZD_ROA_INSTLSTATUS (7C) FEP17 143
SZD_ROA_JOURNALNAME (80) FEP17 143
SZD_ROA_LASTACQCODE (9C) FEP17 143
SZD_ROA_LINES (98) FEP17 143
SZD_ROA_MDT 144
SZD_ROA_MSGJRNL (88) FEP17 143
SZD_ROA_NODE (B0) FEP17 143
SZD_ROA_OUC1 (A0) FEP17 143
SZD_ROA_OUC2 (A8) FEP17 143
SZD_ROA_OUC3 (B0) FEP17 143
SZD_ROA_OUT1 (78) FEP17 142
SZD_ROA_OUT2 (7A) FEP17 143
SZD_ROA_OUT3 (7C) FEP17 143
SZD_ROA_OUT5 143
SZD_ROA_OUT6 (88) FEP17 143
SZD_ROA_OUTLINE (B4) FEP17 144
SZD_ROA_PASSTICKET (A0) FEP17 143
SZD_ROA_POOL (A0) FEP17 143
SZD_ROA_POSITION (98) FEP17 143
SZD_ROA_PROPERTYSET (B0) FEP17 143
SZD_ROA_PROTECT 144
SZD_ROA_PS (B3) FEP17 144
SZD_ROA_REASON (7B) FEP17 143
SZD_ROA_REMFLENGTH (94) FEP17 143
SZD_ROA_RES1 (8C) FEP17 143
SZD_ROA_RES2 (90) FEP17 143
SZD_ROA_RES3 (94) FEP17 143
SZD_ROA_RES4 (98) FEP17 143
SZD_ROA_RES5 (9C) FEP17 143
SZD_ROA_RESPONSE (7A) FEP17 143
SZD_ROA_RESPSTATUS (7A) FEP17 143
SZD_ROA_SENSEDATA (8C) FEP17 143
SZD_ROA_SEQNUMIN (98) FEP17 143
SZD_ROA_SEQNUMOUT (9C) FEP17 143

SZD_ROA_SERVSTATUS (78) FEP17 142
SZD_ROA_SESSNSTATUS (78) FEP17 142
SZD_ROA_SIZE (9C) FEP17 143
SZD_ROA_STATE (80) FEP17 143
SZD_ROA_STSNSTATUS (78) FEP17 142
SZD_ROA_TARGET (A8) FEP17 143
SZD_ROA_TRANSPARENCY (B5) FEP17 144
SZD_ROA_VALIDATION (B2) FEP17 144
SZD_ROA_WAITCONVNUM (98) FEP17 143
SZD_RPA_EYE (0) FEP17 141
SZD_SC_CD (3C) FEP18 145
SZD_SC_CM (38) FEP18 145
SZD_SC_DYNAA (34) FEP18 145
SZD_SC_DYNAL (44) FEP18 145
SZD_SC_EYE (0) FEP18 145
SZD_SC_FLAGS (2C) FEP18 145
SZD_SC_ND (40) FEP18 145
SZD_SC_QEB 145
SZD_SC_QNEXT 145
SZD_SC_REQTYPE 145
SZD_SC_RPL (48) FEP18 145
SZD_SC_TRINTVL (30) FEP18 145
SZD_SC_TRTYPE (32) FEP18 145
SZD_SC_VTAM (48) FEP18 145
SZD_SR_ALLOCATESWAITING (44) FEP19 146
SZD_SR_EYE (0) FEP19 146
SZD_SR_NDPTR (34) FEP19 146
SZD_SR_NEXT (24) FEP19 146
SZD_SR_NODES (3C) FEP19 146
SZD_SR_ORNEXT (2C) FEP19 146
SZD_SR_ORPREV (28) FEP19 146
SZD_SR_PDPTR (30) FEP19 146
SZD_SR_PKALLOCATESWAITING (48) FEP19 146
SZD_SR_PREV 146
SZD_SR_TDPTR (34) FEP19 146
SZD_SR_TIMEOUTS (50) FEP19 146
SZD_SR_TOTALLOCATES (40) FEP19 146
SZD_SR_TOTALLOCATEWAITS (4C) FEP19 146
SZD_SR_USAGE (38) FEP19 146
SZD_STQ_EYE (0) FEP06 123
SZD_STQ_QNEXT 123
SZD_STQ_QREQ (0) FEP06 123
SZD_TCA_SAVE (B4) FEP06 121
SZD_TD_API (48) FEP20 148
SZD_TD_CDLIST (54) FEP20 148
SZD_TD_CS_FLAGS 147
SZD_TD_CURRENT (6C) FEP20 148
SZD_TD_DEFTRAN (3C) FEP20 147
SZD_TD_EYE (0) FEP20 147
SZD_TD_INSTSTATUS (6A) FEP20 148
SZD_TD_NAME (58) FEP20 148
SZD_TD_NEXT (4C) FEP20 148
SZD_TD_ON_Q 147
SZD_TD_ON_QIRB (BIT) FEP20 147
SZD_TD_ON_TMR (BIT) FEP20 147
SZD_TD_PLUN (60) FEP20 148
SZD_TD_PREV (48) FEP20 148
SZD_TD_QC (24) FEP20 147
SZD_TD_QCB (20) FEP20 147
SZD_TD_QP (20) FEP20 147
SZD_TD_RCOUNT (74) FEP20 148
SZD_TD_RE_CTR (44) FEP20 147
SZD_TD_RE_QC (40) FEP20 147
SZD_TD_RE_QCB (40) FEP20 147
SZD_TD_REQ (28) FEP20 147
SZD_TD_REQ_FAIL (BIT) FEP20 147
SZD_TD_SERVSTATUS (68) FEP20 148
SZD_TD_SRLIST (50) FEP20 148
SZD_TD_TRINTVL 147
SZD_TD_TRTYPE (32) FEP20 147
SZD_TD_UDATA (78) FEP20 148
SZD_TD_USAGE (70) FEP20 148
SZD_TD_WE 147
SZD_TDQ_EYE (0) FEP06 122
SZD_TDQ_QNEXT 122
SZD_TDQ_QREQ (0) FEP06 122
SZD_USQ_EYE (0) FEP06 123
SZD_USQ_QNEXT 123
SZD_USQ_QREQ (0) FEP06 123
SZK_ADD_NODE 2 FEP06 125
SZK_ADD_TARGET 2 FEP06 125
SZK_CC_OK 1 FEP06 125
SZK_DS_END 2 FEP06 125
SZK_DS_INIT 2 FEP06 125
SZK_DS_RUN 2 FEP06 125
SZK_DS_WAIT 2 FEP06 125
SZK_FLAG_OFF 0 FEP06 124
SZK_FLAG_ON 0 FEP06 124
SZK_IRB_LENGTH 4 FEP06 124
SZK_LIFO_LENGTH 4 FEP06 124
SZK_RASIZE 4 FEP06 124
SZK_RC_DEFER 4 FEP06 125
SZK_RC_EMPTY 4 FEP06 125
SZK_RC_INVREQ 4 FEP06 125
SZK_RC_NO_STORAGE 4 FEP06 125
SZK_RC_NOPOST 4 FEP06 125
SZK_RC_OK 4 FEP06 125
SZK_RC_POST 4 FEP06 125
SZK_RDN_NODE_DELETED 2 FEP06 125
SZK_REISSUE 4 FEP06 124
SZK_REOPEN 4 FEP06 124
SZK_REQUEUE 4 FEP06 124
SZK_RNC 4 FEP06 124
SZK_RNCT 4 FEP06 124
SZK_RSC 4 FEP06 124
SZK_RSCT 4 FEP06 124
SZK_RTC 4 FEP06 124
SZK_RTCT 4 FEP06 124
SZK_SFAIL_BIND 4 FEP06 124
SZK_SFAIL_CINIT 4 FEP06 124
SZK_SFAIL_PLU 4 FEP06 124
SZK_SFAIL_REQSESS_INHIBITED 4 FEP06 124
SZK_SFAIL_REQSESS_NOT_AVAIL 4 FEP06 124
SZK_SFAIL_REQSESS_OTHER 4 FEP06 124
SZK_SFAIL_SLU 4 FEP06 124
SZK_SFAIL_SSCP 4 FEP06 124
SZK_SFAIL_UNDEF_SETUP 4 FEP06 124
SZK_SLOST_CLEANUP_ABNORM 4 FEP06 124
SZK_SLOST_CLEANUP_NORM 4 FEP06 124
SZK_SLOST_LOSTERM 4 FEP06 124
SZK_SLOST_TAKEDOWN 4 FEP06 124
SZK_SLOST_UNBIND_BIND 4 FEP06 124
SZK_SLOST_UNBIND_INVALID 4 FEP06 124
SZK_SLOST_UNBIND_NORMAL 4 FEP06 124
SZK_SLOST_UNBIND_RECOV 4 FEP06 124
SZK_SLOST_UNBIND_UNRECOV 4 FEP06 124
SZK_SLU2 4 FEP06 124
SZK_SLUP 4 FEP06 124
SZK_TS_TICKLEN 4 FEP06 124
SZS_CONFDATA 148
SZS_SP_AC (40) FEP21 148
SZS_SP_CD (48) FEP21 148
SZS_SP_CM (50) FEP21 148
SZS_SP_CV (58) FEP21 148
SZS_SP_DA (60) FEP21 148
SZS_SP_DS (68) FEP21 148
SZS_SP_DT (70) FEP21 149
SZS_SP_NB (78) FEP21 149
SZS_SP_ND (80) FEP21 149
SZS_SP_PD (88) FEP21 149
SZS_SP_PS (90) FEP21 149
SZS_SP_RP (98) FEP21 149
SZS_SP_RQ (A0) FEP21 149
SZS_SP_SR (B8) FEP21 149
SZS_SP_TD (A8) FEP21 149
SZS_SP_WE (B0) FEP21 149
SZS_SYSSTATE (10) FEP21 148
SZS_SYSSTATE_CLOSED 4 FEP21 149
SZS_SYSSTATE_FAILED 4 FEP21 149
SZS_SYSSTATE_INITING 4 FEP21 149
SZS_SYSSTATE_NEVAC 4 FEP21 149
SZS_SYSSTATE_OPEN 4 FEP21 149
SZS_SYSSTATE_TERM_FORCE 4 FEP21 149
SZS_SYSSTATE_TERM_IMMED 4 FEP21 149
SZS_SYSSTATE_TERM_NORM 4 FEP21 149
SZSANCCI 148
SZSANCRM (24) FEP21 148
SZSEND (140) FEP21 149
SZSEYEC (2) FEP21 148
SZSEYEL (0) FEP21 148
SZSLEN 4 FEP21 149
SZSTLEV (16) FEP21 148
SZSTMODE (14) FEP21 148
SZSTMODE_DYNAMIC 2 FEP21 149
SZSTMODE_QR 2 FEP21 149
SZSTMODE_SZ 2 FEP21 149

T

table
  CICS affinities utility trace table, CAUTR 26
  message table definition, MEMMS 252
  partner table entry, PTE 297
  stack segment table header, LIFO 203
tables
  data tables connection anchor blocks, DTLPS 69
  data tables local access anchor blocks, DTCPS 68
  data tables remote sharing anchor block, DTRPS 72
  data tables security anchor block, DTXPS 74
  data tables SVC routine anchor blocks, DTSPS 72
  file browse work area for data tables, FBWAC 99
TACB_ABEND_CODE 5
TACB_REG_13_AT_ABEND (2B0) APLI 5
TAKE_KEYPOINT (24) RMLI 304
TAKE_KEYPOINT (8C) RMUW 338
TAKE_KEYPOINT (8CC) RMLK 307
TAKESOCKET_PARMS (18) SOA 372
target
  target descriptor, FEP20 147
TARGET (0) BAACT 15
TAS_AR_MODE_ACTIVE (BIT) KECB 157, 158
TAS_ATTACH_TOKEN (24) KECB 156
TAS_BC_PSW (0) KECB 158
TAS_BC_PSW (208) KECB 157
TAS_BC_PSW (2A8) KECB 157
TAS_CICS_DATA (208) KECB 157
TAS_CLOCK_ACTIVE (BIT) KECB 156
TAS_CLOCK_STATUS (9A) KECB 156
TAS_CPU_CLOCK (90) KECB 156
TAS_CURRENT_STACK (18) KECB 156
TAS_CURRENT_STACK_24 (38) KECB 156
TAS_CURRENT_STACK_31 (30) KECB 156
TAS_DEFERRED_ABEND_CODE (B8) KECB 157
TAS_DEFERRED_ABEND_R14_SAVE (B4) KECB 157
TAS_DOMAIN_INDEX (48) KECB 156
TAS_EC_ADD (10) KECB 158
TAS_EC_ADD (218) KECB 157
TAS_EC_ADD (2B8) KECB 158
TAS_EC_BYTE3 157, 158
TAS_EC_PSW (210) KECB 157
TAS_EC_PSW (2B0) KECB 157
TAS_EC_PSW (8) KECB 158
TAS_END_OF_SEGMENT_24 (34) KECB 156
TAS_END_OF_SEGMENT_31 (2C) KECB 156
TAS_ERROR_ACCESS_REG_STORAGE (268) KECB 157
TAS_ERROR_ACCESS_REG_STORAGE (308) KECB 158
TAS_ERROR_ACCESS_REG_STORAGE (60) KECB 158
TAS_ERROR_ACCESS_REGISTERS (268) KECB 157
TAS_ERROR_ACCESS_REGISTERS (308) KECB 158
TAS_ERROR_ACCESS_REGISTERS (60) KECB 158
TAS_ERROR_ADDRESS (1F0) KECB 157
TAS_ERROR_ALET (378) KECB 158
TAS_ERROR_CICS_RB 0 KECB 161
TAS_ERROR_CICS_RB_NOT_ACTIVE (BIT) KECB 157
TAS_ERROR_CODE (1D8) KECB 157
TAS_ERROR_COUNT (46) KECB 156
TAS_ERROR_DATA (0) KECB 158
TAS_ERROR_DUMP_REQUESTED (BIT) KECB 157
TAS_ERROR_EXECUTING_RB (BIT) KECB 157
TAS_ERROR_FP_REG_0 (350) KECB 158
TAS_ERROR_FP_REG_2 (358) KECB 158
TAS_ERROR_FP_REG_4 (360) KECB 158
TAS_ERROR_FP_REG_6 (368) KECB 158
TAS_ERROR_FP_REGS (350) KECB 158
TAS_ERROR_IN_SUBSPACE (BIT) KECB 158
TAS_ERROR_INFORMATION (1D8) KECB 157
TAS_ERROR_IRB (BIT) KECB 157
TAS_ERROR_KEY (1C) KECB 158
TAS_ERROR_KEY (224) KECB 157
TAS_ERROR_KEY (2C4) KECB 158
TAS_ERROR_MVS_FLAGS (1E1) KECB 157
TAS_ERROR_NUMBER (200) KECB 157
TAS_ERROR_OFFSET (1E6) KECB 157
TAS_ERROR_PROGRAM (1E8) KECB 157
TAS_ERROR_REASON (204) KECB 157
TAS_ERROR_REASON_PRESENT 157
TAS_ERROR_REGISTER_STORAGE 157, 158
TAS_ERROR_REGISTERS (20) KECB 158
TAS_ERROR_REGISTERS (228) KECB 157
TAS_ERROR_REGISTERS (2C8) KECB 158
TAS_ERROR_SRB_MODE (BIT) KECB 157
TAS_ERROR_STOKEN (370) KECB 158
TAS_ERROR_SUBSPACE_FLAGS (37C) KECB 158
TAS_ERROR_TIMESTAMP 158
TAS_ERROR_TYPE (1E0) KECB 157
TAS_FREE_SEGS_24 (1C) KECB 156
TAS_FREE_SEGS_31 (8C) KECB 156
TAS_INDEX (C) KECB 156
TAS_INIT_SEG_24 (AC) KECB 157
TAS_INIT_SEG_31 (B0) KECB 157
TAS_INSTRUCTION_ADDRESS (18) KECB 158
TAS_INSTRUCTION_ADDRESS (220) KECB 157
TAS_INSTRUCTION_ADDRESS (2C0) KECB 158
TAS_INT_DATA 157
TAS_KTCB_ENTRY (40) KECB 156
TAS_MONITORING_TOKEN (20) KECB 156
TAS_NAME (0) KECB 156
TAS_NEXT_FREE (8) KECB 156
TAS_NEXT_TASK (A8) KECB 157
TAS_NQ_WORK_TOKEN (BC) KECB 157
TAS_PARAMETER_LIST (D8) KECB 157
TAS_PREV_TASK (A4) KECB 157
TAS_PURGE_PROTECTION_COUNT (9E) KECB 157
TAS_REGISTER_SAVE (4C) KECB 156
TAS_REGISTER_STORAGE (4C) KECB 156
TAS_RUNAWAY_ACTIVE (BIT) KECB 156
TAS_RUNAWAY_EXPIRED (BIT) KECB 156
TAS_RUNAWAY_LEFT (98) KECB 156
TAS_RUNAWAY_STATE_INITIALISED (BIT) KECB 156
TAS_RUNAWAY_STOPPED (BIT) KECB 156
TAS_SEGMENT_ENTRY_24 (14) KECB 156
TAS_SEGMENT_ENTRY_31 (10) KECB 156
TAS_SEGMENT_POINTERS (2C) KECB 156
TAS_STACK_POINTERS (10) KECB 156
TAS_STATE (3C) KECB 156
TAS_STATE_ACQUIRED_FROM_SM (BIT) KECB 156
TAS_STATE_ALLOCATED (BIT) KECB 156
TAS_STATE_DISPOSABLE (BIT) KECB 156
TAS_STATE_DYNAMIC (BIT) KECB 156
TAS_STATE_LINKAGE_ERROR (BIT) KECB 156
TAS_STATE_SPECIAL (BIT) KECB 156
TAS_STATE_STANDARD (BIT) KECB 156
TAS_STATE_SUPPRESSED (BIT) KECB 156
TAS_STATE_TEMP_STATIC (3D) KECB 156
TAS_STOP_RUNAWAY (9C) KECB 156
TAS_SYSTEM_INT 157
TAS_SYSTEM_RUNAWAY (BIT) KECB 156
TAS_TAS_ADDRESS (1FC) KECB 157
TAS_TAS_ATTACH_TOKEN (1F4) KECB 157
TAS_TAS_TCA_ADDRESS (1F8) KECB 157
TAS_TCA_ADDRESS (28) KECB 156
TAS_TCB_ID (C0) KECB 157
TAS_TOTAL_TIME (90) KECB 156
TAS_TRACE_COUNT (44) KECB 156
TAS_USER_INT (1E4) KECB 157
TAS_XM_TRANSACTION_TOKEN (A0) KECB 157
task
  cics/db2 life of task block, D2LOT 93
  dispatcher domain task description, DSTSK 64
  task browse area, DSTBA 63
TASK (0) DSTSK 64
TASK_CELL_ROOT (B0) DSANC 55
TASK_END (E1) DSTSK 66
TASK_ENTRY (0) KECB 156
TASK_MISC_FLAGS (6A) DSTSK 66
TASK_MODE (68) DSTSK 66
TASK_PAGE_MAP (10) DSANC 62
TASK_STATE (44) DSTSK 65
TASKS_IN_BLOCK 4 DSTSK 68
TBB (0) DUFC 75
TBB_DIR_ELEMENT_ADDRESS (4) DUFC 75
TBB_EYECATCHER (0) DUFC 75
TBB_EYECATCHER_VALUE 4 DUFC 76
TBSS_PTR (8) RDAB 299
TCACLASS 1 SMMCC 366
TCB_ANC_ADDR (30) DSANC 59
TCB_AVAILABLE 58
TCB_COUNT (19C) DSANC 57
TCB_COUNT (1C) DSANC 60
TCB_DS_OLD_CPU_TIME (D0) DSANC 59
TCB_DS_TOT_ACC_CPU_TIME (C8) DSANC 59
TCB_ID (9C) DSANC 59
TCB_ID_RANGE (1B0) DSANC 57

TCB_ID_RANGE (30) DSANC 60
TCB_LIST (18) DSANC 60
TCB_LIST (198) DSANC 57
TCB_MODE (4C) DSANC 59
TCB_MODENAME (9C) DSANC 59
TCB_NUMBER (9E) DSANC 59
TCB_OLD_CPU_TIME (C0) DSANC 59
TCB_POSTED (BIT) DSANC 59
TCB_SAVE_ACC_TIME (B0) DSANC 59
TCB_SAVE_WAIT_TIME (A8) DSANC 59
TCB_SAVED_CPU_FIELDS (A8) DSANC 59
TCB_SUBD_NAME (44) DSANC 59
TCB_SUBD_PTR (14) DSANC 58
TCB_TERM_BEFORE_DELETE_TCB (BIT) DSANC 59
TCB_TOTAL_ACC_CPU_TIME (B8) DSANC 59
TCB_WAITING (BIT) DSANC 59
TCBKEY9 (BIT) DSANC 57, 60
TCL_ARROW (2) XMCLC 439
TCL_BLOCK_NAME (8) XMCLC 439
TCL_CURRENT_ACTIVE (48) XMCLC 439
TCL_CURRENT_QUEUED (4C) XMCLC 439
TCL_DEFINED_MAX_ACTIVE (38) XMCLC 439
TCL_DEFINED_PURGE_THRESHOLD (3C) XMCLC 439
TCL_DEFINITION_FLAGS (40) XMCLC 439
TCL_DEFINITION_STATE (38) XMCLC 439
TCL_DFH (3) XMCLC 439
TCL_DOMID (6) XMCLC 439
TCL_DUMMY_ENTRY (BIT) XMCLC 439
TCL_DUMMY_WARNING_MSG_ISSUED (BIT) XMCLC 439
TCL_INSTANCE_NUMBER (2C) XMCLC 439
TCL_LENGTH (0) XMCLC 439
TCL_LOCK_COUNT (20) XMCLC 439
TCL_LOCK_TOKEN (30) XMCLC 439
TCL_MAX_QUEUED (44) XMCLC 439
TCL_NEXT_TCLASS (18) XMCLC 439
TCL_OPERATIONAL_STATE (44) XMCLC 439
TCL_PEAK_ACTIVE (68) XMCLC 439
TCL_PEAK_QUEUED (6C) XMCLC 439
TCL_PREFIX (0) XMCLC 439
TCL_PURGED_IMMEDIATELY (5C) XMCLC 439
TCL_PURGED_WHILE_QUEUING (64) XMCLC 439
TCL_STATISTICS 439
TCL_TCLASS_ADDRESS (28) XMCLC 439
TCL_TCLASS_NAME (10) XMCLC 439
TCL_TCLASS_TOKEN 439
TCL_TIMES_AT_MAX_ACTIVE (70) XMCLC 439
TCL_TIMES_AT_PURGE_THRESHOLD (74) XMCLC 439
TCL_TOTAL_ATTACHES (58) XMCLC 439
TCL_TOTAL_QUEUED (60) XMCLC 439
TCL_TOTAL_QUEUING_TIME (78) XMCLC 439
TCL_TRANSACTION_QUEUE_HEAD (50) XMCLC 439
TCL_USAGE_COUNT (1C) XMCLC 439
TCLASS_CATALOG_RECORD (0) XMCAT 439
TCTTE_PTR (1C) CPCPS 32
TDQ_CONVID (5C) FEP06 123
TDQ_DATATYPE (2C) FEP06 122
TDQ_DEVICE (64) FEP06 123
TDQ_EVENT1 (38) FEP06 122
TDQ_EVENT2 (3C) FEP06 122
TDQ_EVENTDATA (38) FEP06 122
TDQ_EVENTTYPE (30) FEP06 122
TDQ_EVENTVALUE (34) FEP06 122
TDQ_FORMAT (68) FEP06 123
TDQ_NODE (54) FEP06 123
TDQ_POOL (44) FEP06 122
TDQ_QUEUE (74) FEP06 123
TDQ_QUEUER (24) FEP06 122
TDQ_SPARE4 (40) FEP06 122
TDQ_SPARE8 (6C) FEP06 123
TDQ_TARGET (4C) FEP06 123
TDQDATA 122
TEMP_HIGH_PRIORITY (BIT) DSTSK 65
template
  document handler template descriptor, DHTL 43
temporary
  temporary storage anchor block, TSA 380
  temporary storage auxiliary class, TSAUX 384
  temporary storage main class, TSMN 392
  temporary storage model class, TSMN 390
  temporary storage name class, TSNM 393
  temporary storage ownership lock class, TSOL 394
  temporary storage queue class, TSQU 396
temporary *(continued)*
  temporary storage resource lock class, TSRL 401
  temporary storage shared class, TSRL 399
  temporary storage wait queue class, TSWQ 402
TERMCODE (17C) APLI 4
TERMCODE_BIT0 4
TERMCODE_BIT1 (BIT) APLI 4
TERMCODE_BIT10 (BIT) APLI 4
TERMCODE_BIT11 (BIT) APLI 4
TERMCODE_BIT2 (BIT) APLI 4
TERMCODE_BIT3 (BIT) APLI 4
TERMCODE_BIT4 (BIT) APLI 4
TERMCODE_BIT5 (BIT) APLI 4
TERMCODE_BIT6 (BIT) APLI 4
TERMCODE_BIT7 (BIT) APLI 4
TERMCODE_BIT8 4
TERMCODE_BIT9 (BIT) APLI 4
TERMID (0) RMUW 336
TERMID (33) RMLK 311
TERMID (33) RMUW 331
terminal
  terminal simulation facility, FEP19 146
TERMINAL_LUNAME (37) RMLK 311
TERMINAL_LUNAME (37) RMUW 331
TERMINAL_LUNAME (4) RMUW 336
TERMINATED 1 DDCBC 37
TERMINATED 4 SMDCC 362
TERMINATED 4 TSA 381
TERMINATED 4 XMANC 437
TERMINATING 4 MEPS 259
TERMINATING 4 XMANC 437
TERMINFO (17C) APLI 4
TEXT_ELEMENT 1 MEMMS 256
TEXT_STRING 4 MEMMS 256
THREAD_FREE 4 CCGD 31
TIA 378
TIA_ARROW (2) TIA 378
TIA_BLOCK_NAME (8) TIA 378
TIA_DFH (3) TIA 378
TIA_DISPATCHER_TOKEN (1C) TIA 378
TIA_DOMID (6) TIA 378
TIA_FIRST_TRE_PTR (30) TIA 378
TIA_FLAGS (38) TIA 378
TIA_LENGTH (0) TIA 378
TIA_LOCK_TOKEN (10) TIA 378
TIA_NEXT_EXPIRY_HIGH (20) TIA 378
TIA_NEXT_EXPIRY_LOW (24) TIA 378
TIA_NEXT_EXPIRY_TIME (20) TIA 378
TIA_NUDGE_STATUS (18) TIA 378
TIA_PREFIX (0) TIA 378
TIA_REQUEST_COUNTER (34) TIA 378
TIA_SUSPEND_TOKEN (14) TIA 378
TIA_TIMER_AVAILABLE (BIT) TIA 378
TID_EITS_ENTRY 2 TSA 383
TID_EITS_EXIT 2 TSA 383
TID_EITS_INVALID_FORMAT 2 TSA 383
TID_EITS_INVALID_FUNCTION 2 TSA 383
TID_EITS_INVALID_TS_FUNCTION 2 TSA 383
TID_EITS_RECOVERY 2 TSA 383
TID_LGDM_ENTRY 2 LGANC 193
TID_LGDM_EXIT 2 LGANC 193
TID_LGDM_GET_PARAMETERS_FAILED 2 LGANC 193
TID_LGDM_INVALID_EXIT_ID 2 LGANC 193
TID_LGDM_INVALID_FORMAT 2 LGANC 193
TID_LGDM_INVALID_FUNCTION 2 LGANC 193
TID_LGDM_NO_STORAGE_FOR_LGA 2 LGANC 193
TID_LGDM_RECOVERY 2 LGANC 193
TID_LGDM_REGISTER_ERROR 2 LGANC 193
TID_LGDM_RELEASE_LGUOW_ERROR 2 LGANC 193
TID_LGDM_RELEASE_LOCK_ERROR 2 LGANC 193
TID_LGDM_SET_GATE_ERROR 2 LGANC 193
TID_LGGL_ADD_SUBPOOL_ERROR 2 LGANC 194
TID_LGGL_ADD_UW_SUBPOOL_ERROR 2 LGANC 194
TID_LGGL_BAD_LOGTYPE 2 LGANC 194
TID_LGGL_END_WT_BROWSE_ERROR 2 LGANC 194
TID_LGGL_ENTRY 2 LGANC 193
TID_LGGL_EXIT 2 LGANC 193
TID_LGGL_GET_EXC_LGUOW_LOCK_ERROR 2 LGANC 194
TID_LGGL_GET_EXC_LOCK_ERROR 2 LGANC 193
TID_LGGL_GET_NEXT_WT_ERROR 2 LGANC 194
TID_LGGL_GET_SHR_LOCK_ERROR 2 LGANC 193
TID_LGGL_GET_SHR_SMF_LOCK_ERROR 2 LGANC 194
TID_LGGL_GET_SHR_STREAM_LOCK_ERROR 2 LGANC 194

TID_LGGL_GLOGS_BBLX_EXCEPTION 2 LGANC 194
TID_LGGL_GLOGS_SIF_EXCEPTION 2 LGANC 194
TID_LGGL_INVALID_FORMAT 2 LGANC 193
TID_LGGL_INVALID_FUNCTION 2 LGANC 193
TID_LGGL_INVALID_PARAMETERS 2 LGANC 194
TID_LGGL_MVS_FORCE_ERROR 2 LGANC 194
TID_LGGL_MVS_WRITE_ERROR 2 LGANC 194
TID_LGGL_REC_RLSE_LGUOW_LOCK_ERROR 2 LGANC 194
TID_LGGL_REC_RLSE_SMF_LOCK_ERROR 2 LGANC 194
TID_LGGL_REC_RLSE_STREAM_LOCK_ERROR 2 LGANC 194
TID_LGGL_RECOVERY 2 LGANC 193
TID_LGGL_RECOVERY_RELEASE_LOCK_ERROR 2 LGANC 194
TID_LGGL_RELEASE_EXC_LGUOW_LOCK_ERROR 2 LGANC 194
TID_LGGL_RELEASE_EXC_LOCK_ERROR 2 LGANC 193
TID_LGGL_RELEASE_SHR_LOCK_ERROR 2 LGANC 193
TID_LGGL_RELEASE_SHR_SMF_LOCK_ERROR 2 LGANC 194
TID_LGGL_RELEASE_SHR_STREAM_LOCK_ERROR 2 LGANC 194
TID_LGGL_SMF_FORCE_ERROR 2 LGANC 194
TID_LGGL_SMF_WRITE_ERROR 2 LGANC 194
TID_LGGL_START_WT_BROWSE_ERROR 2 LGANC 194
TID_LGGL_STORAGE_REQ_PURGED 2 LGANC 194
TID_LGGL_UNKNOWN_KE_ERROR_CODE 2 LGANC 193
TID_LGGL_UNKNOWN_LOG_TOKEN 2 LGANC 194
TID_LGJN_ADD_ENQPOOL_ERROR 2 LGANC 195
TID_LGJN_ADD_SUBPOOL_ERROR 2 LGANC 195
TID_LGJN_BROWSES_BBLX_EXCEPTION 2 LGANC 195
TID_LGJN_BROWSES_SIF_EXCEPTION 2 LGANC 195
TID_LGJN_CATLG_DELETE_ERROR 2 LGANC 195
TID_LGJN_CATLG_WRITE_ERROR 2 LGANC 195
TID_LGJN_DEQUEUE_ERROR 2 LGANC 195
TID_LGJN_ENQUEUE_ERROR 2 LGANC 195
TID_LGJN_ENTRY 2 LGANC 194
TID_LGJN_EXIT 2 LGANC 194
TID_LGJN_GET_EXC_LOCK_ERROR 2 LGANC 194
TID_LGJN_GET_EXC_SMF_LOCK_ERROR 2 LGANC 195
TID_LGJN_GET_SHR_LOCK_ERROR 2 LGANC 194
TID_LGJN_GET_SHR_SMF_LOCK_ERROR 2 LGANC 195
TID_LGJN_GET_SHR_STREAM_LOCK_ERROR 2 LGANC 195
TID_LGJN_INVALID_FORMAT 2 LGANC 194
TID_LGJN_INVALID_FUNCTION 2 LGANC 194
TID_LGJN_INVALID_JNL_STATUS 2 LGANC 195
TID_LGJN_INVALID_SET_STATUS 2 LGANC 195
TID_LGJN_JNL_CONN_ERROR 2 LGANC 195
TID_LGJN_JNL_DEFINED 2 LGANC 195
TID_LGJN_JNL_DISCARDED 2 LGANC 195
TID_LGJN_JOURNALS_BBLX_EXCEPTION 2 LGANC 195
TID_LGJN_JOURNALS_SIF_EXCEPTION 2 LGANC 195
TID_LGJN_LD_MATCH_ERROR 2 LGANC 195
TID_LGJN_REC_RLSE_SMF_LOCK_ERROR 2 LGANC 195
TID_LGJN_REC_RLSE_STREAM_LOCK_ERROR 2 LGANC 195
TID_LGJN_RECOVERY 2 LGANC 194
TID_LGJN_RECOVERY_RELEASE_LOCK_ERROR 2 LGANC 195
TID_LGJN_RELEASE_EXC_LOCK_ERROR 2 LGANC 194
TID_LGJN_RELEASE_EXC_SMF_LOCK_ERROR 2 LGANC 195
TID_LGJN_RELEASE_SHR_LOCK_ERROR 2 LGANC 194
TID_LGJN_SMF_CONN_ERROR 2 LGANC 195
TID_LGJN_STREAM_FAILED 2 LGANC 195
TID_LGJN_UNKNOWN_KE_ERROR_CODE 2 LGANC 194
TID_LGLD_ADD_SUBPOOL_ERROR 2 LGANC 195
TID_LGLD_BROWSES_BBLX_EXCEPTION 2 LGANC 196
TID_LGLD_BROWSES_SIF_EXCEPTION 2 LGANC 196
TID_LGLD_CATLG_DELETE_ERROR 2 LGANC 196
TID_LGLD_CATLG_WRITE_ERROR 2 LGANC 196
TID_LGLD_ENTRY 2 LGANC 195
TID_LGLD_EXIT 2 LGANC 195
TID_LGLD_GET_EXC_LOCK_ERROR 2 LGANC 195
TID_LGLD_GET_SHR_LOCK_ERROR 2 LGANC 195
TID_LGLD_INVALID_FORMAT 2 LGANC 195
TID_LGLD_INVALID_FUNCTION 2 LGANC 195
TID_LGLD_JOURNALMODEL_DISCARDED 2 LGANC 196
TID_LGLD_JOURNALMODEL_INSTALLED 2 LGANC 196
TID_LGLD_JOURNALMODEL_REPLACED 2 LGANC 196
TID_LGLD_JOURNALMODELS_BBLX_EXCEPTION 2 LGANC 196
TID_LGLD_JOURNALMODELS_SIF_EXCEPTION 2 LGANC 196
TID_LGLD_RECOVERY 2 LGANC 195
TID_LGLD_RECOVERY_RELEASE_LOCK_ERROR 2 LGANC 195
TID_LGLD_RELEASE_EXC_LOCK_ERROR 2 LGANC 195
TID_LGLD_RELEASE_SHR_LOCK_ERROR 2 LGANC 195
TID_LGLD_UNKNOWN_KE_ERROR_CODE 2 LGANC 195
TID_LGPA_ENTRY 2 LGANC 197
TID_LGPA_EXIT 2 LGANC 197
TID_LGPA_INVALID_FORMAT 2 LGANC 197
TID_LGPA_INVALID_FUNCTION 2 LGANC 197
TID_LGPA_RECOVERY 2 LGANC 197
TID_LGSC_ENTRY 2 LGANC 197
TID_LGSC_EXIT 2 LGANC 197
TID_LGSC_INVALID_FORMAT 2 LGANC 197
TID_LGSC_INVALID_FUNCTION 2 LGANC 197
TID_LGSC_INVALID_PARMS 2 LGANC 197
TID_LGSC_RECOVERY 2 LGANC 197
TID_LGST_ADD_BROWSES_SUBPOOL_ERROR 2 LGANC 196
TID_LGST_ADD_ENQPOOL_ERROR 2 LGANC 196
TID_LGST_ADD_STREAM_LOCK_ERROR 2 LGANC 197
TID_LGST_ADD_SUBPOOL_ERROR 2 LGANC 196
TID_LGST_BROWSES_BBLX_EXCEPTION 2 LGANC 196
TID_LGST_BROWSES_SIF_EXCEPTION 2 LGANC 196
TID_LGST_CONNECT_ERROR 2 LGANC 197
TID_LGST_DEQUEUE_ERROR 2 LGANC 196
TID_LGST_END_WT_BROWSE_ERROR 2 LGANC 197
TID_LGST_ENQUEUE_ERROR 2 LGANC 196
TID_LGST_ENTRY 2 LGANC 196
TID_LGST_EXIT 2 LGANC 196
TID_LGST_EXIT_REJECTED_DEFINE 2 LGANC 197
TID_LGST_GET_COND_STREAM_LOCK_ERROR 2 LGANC 196
TID_LGST_GET_EXC_LGUOW_LOCK_ERROR 2 LGANC 197
TID_LGST_GET_EXC_LOCK_ERROR 2 LGANC 196
TID_LGST_GET_EXC_STREAM_LOCK_ERROR 2 LGANC 196
TID_LGST_GET_NEXT_WT_ERROR 2 LGANC 197
TID_LGST_GET_SHR_LOCK_ERROR 2 LGANC 196
TID_LGST_INVALID_FORMAT 2 LGANC 196
TID_LGST_INVALID_FUNCTION 2 LGANC 196
TID_LGST_MVS_DEQ_FAIL 2 LGANC 197
TID_LGST_MVS_DEQ_INPUT 2 LGANC 197
TID_LGST_MVS_DEQ_OK 2 LGANC 197
TID_LGST_MVS_ENQ_FAIL 2 LGANC 197
TID_LGST_MVS_ENQ_INPUT 2 LGANC 197
TID_LGST_MVS_ENQ_OK 2 LGANC 197
TID_LGST_REC_RLSE_LGUOW_LOCK_ERROR 2 LGANC 197
TID_LGST_REC_RLSE_STREAM_LOCK_ERROR 2 LGANC 197
TID_LGST_RECOVERY 2 LGANC 196
TID_LGST_RECOVERY_RELEASE_LOCK_ERROR 2 LGANC 196
TID_LGST_RELEASE_EXC_LGUOW_LOCK_ERROR 2 LGANC 197
TID_LGST_RELEASE_EXC_LOCK_ERROR 2 LGANC 196
TID_LGST_RELEASE_EXC_STREAM_LOCK_ERROR 2 LGANC 196
TID_LGST_RELEASE_SHR_LOCK_ERROR 2 LGANC 196
TID_LGST_RELEASE_SHR_STREAM_LOCK_ERROR 2 LGANC 196
TID_LGST_START_WT_BROWSE_ERROR 2 LGANC 197
TID_LGST_STREAM_DEFINE_ERROR 2 LGANC 196
TID_LGST_STREAM_DEFINE_INPUT 2 LGANC 196
TID_LGST_STREAM_DEFINED 2 LGANC 196
TID_LGST_STREAMS_BBLX_EXCEPTION 2 LGANC 196
TID_LGST_STREAMS_SIF_EXCEPTION 2 LGANC 196
TID_LGST_UNKNOWN_KE_ERROR_CODE 2 LGANC 196
TID_LGST_WAIT_FOR_STREAM_LOCK 2 LGANC 197
TID_SMAD_ENTRY 2 SMDCC 357
TID_SMAD_EXIT 2 SMDCC 357
TID_SMAD_INVALID_FORMAT 2 SMDCC 357
TID_SMAD_INVALID_FUNCTION 2 SMDCC 357
TID_SMAD_INVALID_SUBPOOL_TOKEN 2 SMDCC 357
TID_SMAD_NO_MVS_STORAGE 2 SMDCC 357
TID_SMAD_RECOVERY 2 SMDCC 357
TID_SMAD_SUBPOOL_NOT_EMPTY 2 SMDCC 357
TID_SMAR_ENTRY 2 SMDCC 357
TID_SMAR_EXIT 2 SMDCC 357
TID_SMAR_FREEMAIN_ELEM 2 SMDCC 357
TID_SMAR_INQ_TRAN_FAIL 2 SMDCC 357
TID_SMAR_INQ_TRAN_TOKEN_FAIL 2 SMDCC 357
TID_SMAR_INVALID_FORMAT 2 SMDCC 357
TID_SMAR_INVALID_FUNCTION 2 SMDCC 357
TID_SMAR_NO_MVS_STORAGE_SCA 2 SMDCC 357
TID_SMAR_NO_MVS_STORAGE_SCQ 2 SMDCC 357
TID_SMAR_NO_MVS_STORAGE_SMX 2 SMDCC 357
TID_SMAR_RECOVERY 2 SMDCC 357
TID_SMAR_SET_TRAN_TOKEN_FAIL 2 SMDCC 357
TID_SMAR_STG_VIOL_PCT_INC_FAIL 2 SMDCC 357
TID_SMAR_STG_VIOL_TCT_INC_FAIL 2 SMDCC 357
TID_SMAR_STGCHK_FAILURE 2 SMDCC 357
TID_SMCK_DUP_SAA_NOT_IN_DSA 2 SMDCC 358
TID_SMCK_ENTRY 2 SMDCC 358
TID_SMCK_EXIT 2 SMDCC 358
TID_SMCK_INVALID_FORMAT 2 SMDCC 358
TID_SMCK_INVALID_FUNCTION 2 SMDCC 358
TID_SMCK_LOCK_ERROR 2 SMDCC 358
TID_SMCK_RECOVERY 2 SMDCC 358
TID_SMCK_SAA_CLASS_INVALID 2 SMDCC 358
TID_SMCK_SAA_INV_SUBPOOL_ID 2 SMDCC 358

TID_SMCK_SAA_LENGTH_INVALID 2 SMDCC 358
TID_SMCK_SAA_LENGTH_NOT_MULT8 2 SMDCC 358
TID_SMCK_SAA_LENGTH_ZERO 2 SMDCC 358
TID_SMCK_SAA_NOT_BDY8 2 SMDCC 358
TID_SMCK_SAA_NOT_IN_DSA 2 SMDCC 358
TID_SMCK_SAA_RECOVERED 2 SMDCC 358
TID_SMCK_SAACHK_TP 2 SMDCC 358
TID_SMCK_STG_VIOL_PCT_INC_FAIL 2 SMDCC 359
TID_SMCK_STG_VIOL_TCT_INC_FAIL 2 SMDCC 359
TID_SMCK_SWITCH_FROM_QR_FAIL 2 SMDCC 359
TID_SMCK_SWITCH_TO_QR_FAIL 2 SMDCC 359
TID_SMCK_TCTTE_RECOVERED 2 SMDCC 358
TID_SMCK_TIOA_CHAIN_LOOP 2 SMDCC 359
TID_SMCK_UNLOCK_ERROR 2 SMDCC 358
TID_SMCK_ZONE_CHECK_FAILED 2 SMDCC 359
TID_SMCK_ZONES_RECOVERED 2 SMDCC 359
TID_SMDM_ENTRY 2 SMDCC 356
TID_SMDM_EXIT 2 SMDCC 357
TID_SMDM_INVALID_FORMAT 2 SMDCC 357
TID_SMDM_INVALID_FUNCTION 2 SMDCC 357
TID_SMDM_NOSTG_DFT_DSALIM 2 SMDCC 357
TID_SMDM_NOSTG_DFT_EDSALIM 2 SMDCC 357
TID_SMDM_NOSTG_DSA 2 SMDCC 357
TID_SMDM_NOSTG_REQ_DSALIM 2 SMDCC 357
TID_SMDM_NOSTG_REQ_EDSALIM 2 SMDCC 357
TID_SMDM_NOSTG_SCAB 2 SMDCC 357
TID_SMDM_NOSTG_SCQB 2 SMDCC 357
TID_SMDM_NOSTG_SMA 2 SMDCC 357
TID_SMDM_NOSTG_SMXB 2 SMDCC 357
TID_SMDM_NOSTG_STAB 2 SMDCC 357
TID_SMDM_RECOVERY 2 SMDCC 357
TID_SMDM_STCK_ERROR 2 SMDCC 357
TID_SMDM_SVC_CALL_FAIL 2 SMDCC 357
TID_SMGF_ENTRY 2 SMDCC 357
TID_SMGF_EXIT 2 SMDCC 357
TID_SMGF_FREEMAIN_INV_STG_CLASS 2 SMDCC 358
TID_SMGF_FREEMAIN_NO_TRAN_ENV 2 SMDCC 358
TID_SMGF_GETMAIN_INV_STG_CLASS 2 SMDCC 357
TID_SMGF_GETMAIN_NO_TRAN_ENV 2 SMDCC 358
TID_SMGF_INSUFFICIENT_STORAGE 2 SMDCC 357
TID_SMGF_INV_ADDR_STG_CLASS 2 SMDCC 358
TID_SMGF_INVALID_ADDRESS 2 SMDCC 357
TID_SMGF_INVALID_FUNCTION 2 SMDCC 357
TID_SMGF_INVALID_INITIAL_IMAGE 2 SMDCC 357
TID_SMGF_NEXT_SCF_OVERLAY 2 SMDCC 358
TID_SMGF_NO_MVS_STORAGE 2 SMDCC 357
TID_SMGF_NO_MVS_STORAGE_SQE 2 SMDCC 358
TID_SMGF_PAGES_NOT_OWNED 2 SMDCC 358
TID_SMGF_PREV_SCF_OVERLAY 2 SMDCC 358
TID_SMGF_QCELL_ALREADY_FREE 2 SMDCC 357
TID_SMGF_QCELL_FREEMAIN_INV_QPH 2 SMDCC 357
TID_SMGF_QCELL_GETMAIN_INV_QPF 2 SMDCC 357
TID_SMGF_QCELL_INV_FREE_CHAIN 2 SMDCC 357
TID_SMGF_QCELL_SCAP_FOUND 2 SMDCC 358
TID_SMGF_RECOVERY 2 SMDCC 357
TID_SMGF_STG_FREEZE 2 SMDCC 358
TID_SMGF_STG_VIOL_PCT_INC_FAIL 2 SMDCC 358
TID_SMGF_STG_VIOL_TCT_INC_FAIL 2 SMDCC 358
TID_SMGF_STGCHK_FAILURE 2 SMDCC 357
TID_SMGF_SUBPOOL_LOCK_FAILED 2 SMDCC 358
TID_SMGF_SUBPOOL_UNLOCK_FAILED 2 SMDCC 358
TID_SMMC2_ENTRY 2 SMDCC 360
TID_SMMC2_EXIT 2 SMDCC 360
TID_SMMC2_FREEMAIN_ELEM 2 SMDCC 360
TID_SMMC2_INVALID_ADDRESS 2 SMDCC 360
TID_SMMC2_INVALID_FUNCTION 2 SMDCC 360
TID_SMMC2_NEXT_SCF_OVERLAY 2 SMDCC 360
TID_SMMC2_NO_MVS_STORAGE 2 SMDCC 360
TID_SMMC2_NO_TRAN_ENV 2 SMDCC 360
TID_SMMC2_PAGES_NOT_OWNED 2 SMDCC 360
TID_SMMC2_PREV_SCF_OVERLAY 2 SMDCC 360
TID_SMMC2_RECOVERY 2 SMDCC 360
TID_SMMC2_SAACHK_F_ALL_TP 2 SMDCC 360
TID_SMMC2_STG_VIOL_PCT_INC_FAIL 2 SMDCC 360
TID_SMMC2_STG_VIOL_TCT_INC_FAIL 2 SMDCC 360
TID_SMMC2_STGCHK_FAILURE 2 SMDCC 360
TID_SMMCI_ENTRY 2 SMDCC 358
TID_SMMCI_EXIT 2 SMDCC 358
TID_SMMCI_RECOVERY 2 SMDCC 358
TID_SMMF_ADDR_IN_FREE_PAGE 2 SMDCC 359
TID_SMMF_ADDR_NOT_BDY8 2 SMDCC 359
TID_SMMF_ADDR_OUTSIDE_DSA 2 SMDCC 359
TID_SMMF_ENTRY 2 SMDCC 359
TID_SMMF_EXIT 2 SMDCC 359
TID_SMMF_INVALID_ADDRESS 2 SMDCC 359
TID_SMMF_INVALID_EXEC_KEY 2 SMDCC 360
TID_SMMF_INVALID_FUNCTION 2 SMDCC 360
TID_SMMF_NEXT_SCF_OVERLAY 2 SMDCC 360
TID_SMMF_NO_MVS_STORAGE 2 SMDCC 359
TID_SMMF_NO_TCTTE_ADDRESS 2 SMDCC 359
TID_SMMF_NO_TRAN_ENV 2 SMDCC 360
TID_SMMF_PAGES_NOT_OWNED 2 SMDCC 360
TID_SMMF_PREV_SCF_OVERLAY 2 SMDCC 360
TID_SMMF_RECOVERY 2 SMDCC 359
TID_SMMF_SAACHK_F_TP 2 SMDCC 359
TID_SMMF_STG_FREEZE 2 SMDCC 360
TID_SMMF_STG_VIOL_PCT_INC_FAIL 2 SMDCC 360
TID_SMMF_STG_VIOL_TCT_INC_FAIL 2 SMDCC 360
TID_SMMF_STGCHK_FAILURE 2 SMDCC 360
TID_SMMF_TP_ADDR_NOT_FOUND 2 SMDCC 359
TID_SMMG_CICS24_INV_GET_LENGTH 2 SMDCC 359
TID_SMMG_CICS24_SAA_INV_GET_LEN 2 SMDCC 359
TID_SMMG_CICS31_INV_GET_LENGTH 2 SMDCC 359
TID_SMMG_ENTRY 2 SMDCC 359
TID_SMMG_EXIT 2 SMDCC 359
TID_SMMG_INSUFFICIENT_STORAGE 2 SMDCC 359
TID_SMMG_INV_STORAGE_CLASS 2 SMDCC 359
TID_SMMG_INVALID_FUNCTION 2 SMDCC 359
TID_SMMG_NO_MVS_STORAGE 2 SMDCC 359
TID_SMMG_NO_TCTTE_ADDRESS 2 SMDCC 359
TID_SMMG_NO_TRAN_ENV 2 SMDCC 359
TID_SMMG_RECOVERY 2 SMDCC 359
TID_SMMG_SHRC24_INV_GET_LENGTH 2 SMDCC 359
TID_SMMG_SHRC24_SAA_INV_GET_LEN 2 SMDCC 359
TID_SMMG_SHRC31_INV_GET_LENGTH 2 SMDCC 359
TID_SMMG_SHRU24_INV_GET_LENGTH 2 SMDCC 359
TID_SMMG_SHRU31_INV_GET_LENGTH 2 SMDCC 359
TID_SMMG_TASK_INV_GET_LENGTH 2 SMDCC 359
TID_SMMG_TASK24_INV_GET_LENGTH 2 SMDCC 359
TID_SMMG_TP_INV_GET_LENGTH 2 SMDCC 359
TID_SMMG_USER24_INV_GET_LENGTH 2 SMDCC 359
TID_SMMG_USER31_INV_GET_LENGTH 2 SMDCC 359
TID_SMPP_AFTER_SVC_CALL 2 SMDCC 360
TID_SMPP_ALLOCATE_EXTENT_FAILED 2 SMDCC 360
TID_SMPP_BEFORE_SVC_CALL 2 SMDCC 360
TID_SMPP_DELETING_EMPTY_EXTENT 2 SMDCC 360
TID_SMPP_ENTRY 2 SMDCC 360
TID_SMPP_EXIT 2 SMDCC 360
TID_SMPP_FREE_DSA_LIMIT_FAILED 2 SMDCC 360
TID_SMPP_INVALID_FORMAT 2 SMDCC 360
TID_SMPP_INVALID_FUNCTION 2 SMDCC 360
TID_SMPP_NOSTG_CTN 2 SMDCC 360
TID_SMPP_NOSTG_PPA 2 SMDCC 360
TID_SMPP_NOSTG_PPX 2 SMDCC 360
TID_SMPP_NOSTG_SAT 2 SMDCC 360
TID_SMPP_RECOVERY 2 SMDCC 360
TID_SMPP_SVC_CALL_FAIL 2 SMDCC 360
TID_SMPQ_AFTER_SVC_CALL 2 SMDCC 361
TID_SMPQ_BEFORE_SVC_CALL 2 SMDCC 361
TID_SMPQ_ENTRY 2 SMDCC 360
TID_SMPQ_EXIT 2 SMDCC 360
TID_SMPQ_INSUFFICIENT_STORAGE 2 SMDCC 361
TID_SMPQ_INVALID_ADDRESS 2 SMDCC 361
TID_SMPQ_INVALID_FORMAT 2 SMDCC 361
TID_SMPQ_INVALID_FUNCTION 2 SMDCC 361
TID_SMPQ_NOSTG_CTN 2 SMDCC 361
TID_SMPQ_RECOVERY 2 SMDCC 361
TID_SMPQ_SVC_CALL_FAIL 2 SMDCC 361
TID_SMSCP_ENTRY 2 SMDCC 361
TID_SMSCP_EXIT 2 SMDCC 362
TID_SMSCP_INVALID_REQUEST 2 SMDCC 362
TID_SMSQ_AFTER_SUSPEND 2 SMDCC 360
TID_SMSQ_BEFORE_SUSPEND 2 SMDCC 360
TID_SMSQ_DSSR_INQUIRE_SUSPEND 2 SMDCC 360
TID_SMSQ_ENTRY 2 SMDCC 360
TID_SMSQ_EXIT 2 SMDCC 360
TID_SMSQ_INVALID_FORMAT 2 SMDCC 360
TID_SMSQ_INVALID_FUNCTION 2 SMDCC 360
TID_SMSQ_NO_MVS_STORAGE_SQE 2 SMDCC 360
TID_SMSQ_RECOVERY 2 SMDCC 360
TID_SMSR_ENTRY 2 SMDCC 358
TID_SMSR_EXIT 2 SMDCC 358
TID_SMSR_INVALID_FORMAT 2 SMDCC 358
TID_SMSR_INVALID_FUNCTION 2 SMDCC 358
TID_SMSR_LOCK_ERROR 2 SMDCC 358
TID_SMSR_RECOVERY 2 SMDCC 358

TID_SMSR_UNLOCK_ERROR 2 SMDCC 358
TID_SMST_ENTRY 2 SMDCC 359
TID_SMST_EXIT 2 SMDCC 359
TID_SMST_INVALID_BUFFER 2 SMDCC 359
TID_SMST_INVALID_FORMAT 2 SMDCC 359
TID_SMST_INVALID_FUNCTION 2 SMDCC 359
TID_SMST_INVALID_PARAMETERS 2 SMDCC 359
TID_SMST_LOCK_ERROR 2 SMDCC 359
TID_SMST_RECOVERY 2 SMDCC 359
TID_SMST_UNLOCK_ERROR 2 SMDCC 359
TID_SMSU_ALESERV_ADD_FAIL_ALLOC 2 SMDCC 361
TID_SMSU_ALESERV_ADD_FAIL_STEAL 2 SMDCC 361
TID_SMSU_ALESERV_DELETE_FAIL 2 SMDCC 361
TID_SMSU_ALET_STEAL 2 SMDCC 361
TID_SMSU_ASSIGN_ENTRY 2 SMDCC 361
TID_SMSU_ASSIGN_EXIT 2 SMDCC 361
TID_SMSU_ASSIGN_FAIL_ABEND 2 SMDCC 361
TID_SMSU_BAD_ELEM_ALIGN 2 SMDCC 361
TID_SMSU_BAD_PAGE_MULTIPLE 2 SMDCC 361
TID_SMSU_CHANGE_MODE_FAIL1 2 SMDCC 361
TID_SMSU_CHANGE_MODE_FAIL2 2 SMDCC 361
TID_SMSU_CREATE_SUBSPACE_ENTRY 2 SMDCC 361
TID_SMSU_CREATE_SUBSPACE_EXIT 2 SMDCC 361
TID_SMSU_DELETE_SUBSPACE_ENTRY 2 SMDCC 361
TID_SMSU_DELETE_SUBSPACE_EXIT 2 SMDCC 361
TID_SMSU_ENTRY 2 SMDCC 361
TID_SMSU_EXIT 2 SMDCC 361
TID_SMSU_FREE_SUBSP_TCBS_FAIL 2 SMDCC 361
TID_SMSU_IARSUBSP_ASSIGN_FAIL 2 SMDCC 361
TID_SMSU_IARSUBSP_CREATE_FAIL 2 SMDCC 361
TID_SMSU_IARSUBSP_DELETE_FAIL 2 SMDCC 361
TID_SMSU_IARSUBSP_UNASSIGN_FAIL 2 SMDCC 361
TID_SMSU_INVALID_FORMAT 2 SMDCC 361
TID_SMSU_INVALID_FUNCTION 2 SMDCC 361
TID_SMSU_INVALID_INPUT_SPACE 2 SMDCC 361
TID_SMSU_MULT_UNASSIGN_ENTRY 2 SMDCC 361
TID_SMSU_NO_ALET_TO_STEAL 2 SMDCC 361
TID_SMSU_RECOVERY 2 SMDCC 361
TID_SMSU_SUA_MVS_GETMAIN_FAIL 2 SMDCC 361
TID_SMSU_SVC_CALL_FAIL 2 SMDCC 361
TID_SMSU_TEST 2 SMDCC 361
TID_SMSU_UNASSIGN_ENTRY 2 SMDCC 361
TID_SMSU_UNASSIGN_EXIT 2 SMDCC 361
TID_SMSU_UNASSIGN_FAIL_ABEND 2 SMDCC 361
TID_SMSU_WRONG_TCB_FOR_ALLOCATE 2 SMDCC 361
TID_SMSU_WRONG_TCB_FOR_DELETE 2 SMDCC 361
TID_SMSU_WRONG_TCB_FOR_RELEASE 2 SMDCC 361
TID_SMSY_AFTER_RESUME 2 SMDCC 358
TID_SMSY_BEFORE_SUSPEND 2 SMDCC 358
TID_SMSY_ENTRY 2 SMDCC 358
TID_SMSY_EXIT 2 SMDCC 358
TID_SMSY_INVALID_FORMAT 2 SMDCC 358
TID_SMSY_INVALID_FUNCTION 2 SMDCC 358
TID_SMSY_INVALID_STATE 2 SMDCC 358
TID_SMSY_NOT_SOS 2 SMDCC 358
TID_SMSY_RECOVERY 2 SMDCC 358
TID_SMSY_SOS 2 SMDCC 358
TID_TSAD_ENTRY 2 TSA 383
TID_TSAD_EXIT 2 TSA 383
TID_TSAD_INVALID_FORMAT 2 TSA 383
TID_TSAD_INVALID_FUNCTION 2 TSA 383
TID_TSAD_RECOVERY 2 TSA 383
TID_TSAD_UNLOCK_ERROR_RECOVERY 2 TSA 383
TID_TSAM_1310_ABEND_1 2 TSA 383
TID_TSAM_1310_ABEND_10 2 TSA 383
TID_TSAM_1310_ABEND_11 2 TSA 383
TID_TSAM_1310_ABEND_2 2 TSA 383
TID_TSAM_1310_ABEND_3 2 TSA 383
TID_TSAM_1310_ABEND_4 2 TSA 383
TID_TSAM_1310_ABEND_5 2 TSA 383
TID_TSAM_1310_ABEND_6 2 TSA 383
TID_TSAM_1310_ABEND_7 2 TSA 383
TID_TSAM_1310_ABEND_8 2 TSA 383
TID_TSAM_1310_ABEND_9 2 TSA 383
TID_TSAM_ENTRY 2 TSA 383
TID_TSAM_EXIT 2 TSA 383
TID_TSAM_INVALID_FORMAT 2 TSA 383
TID_TSAM_INVALID_FUNCTION 2 TSA 383
TID_TSAM_RECOVERY 2 TSA 383
TID_TSBR_ENTRY 2 TSA 382
TID_TSBR_EXIT 2 TSA 382
TID_TSBR_INVALID_FORMAT 2 TSA 382
TID_TSBR_INVALID_FUNCTION 2 TSA 382
TID_TSBR_RECOVERY 2 TSA 382
TID_TSBR_UNLOCK_ERROR_RECOVERY 2 TSA 382
TID_TSDM_ENTRY 2 TSA 381
TID_TSDM_EXIT 2 TSA 381
TID_TSDM_INVALID_FORMAT 2 TSA 381
TID_TSDM_INVALID_FUNCTION 2 TSA 381
TID_TSDM_RECOVERY 2 TSA 381
TID_TSDQ_ENTRY 2 TSA 383
TID_TSDQ_ERROR 2 TSA 383
TID_TSDQ_EXIT 2 TSA 383
TID_TSMB_ENTRY 2 TSA 383
TID_TSMB_EXIT 2 TSA 383
TID_TSMB_INVALID_FORMAT 2 TSA 383
TID_TSMB_INVALID_FUNCTION 2 TSA 383
TID_TSMB_RECOVERY 2 TSA 383
TID_TSMB_UNLOCK_ERROR_RECOVERY 2 TSA 383
TID_TSP_ENTRY 2 TSA 383
TID_TSP_EXIT 2 TSA 383
TID_TSP_INVALID_REQUEST 2 TSA 383
TID_TSPT_ENTRY 2 TSA 382
TID_TSPT_EXIT 2 TSA 382
TID_TSPT_INVALID_FORMAT 2 TSA 382
TID_TSPT_INVALID_FUNCTION 2 TSA 382
TID_TSPT_RECOVERY 2 TSA 382
TID_TSPT_UNLOCK_ERROR_RECOVERY 2 TSA 382
TID_TSQR_ENTRY 2 TSA 381
TID_TSQR_EXIT 2 TSA 381
TID_TSQR_INVALID_FORMAT 2 TSA 381
TID_TSQR_INVALID_FUNCTION 2 TSA 382
TID_TSQR_RECOVERY 2 TSA 381
TID_TSQR_UNLOCK_ERROR_RECOVERY 2 TSA 382
TID_TSRM_ENTRY 2 TSA 382
TID_TSRM_EXIT 2 TSA 382
TID_TSRM_INV_INDOUBT_OPERATION 2 TSA 382
TID_TSRM_INVALID_FORMAT 2 TSA 382
TID_TSRM_INVALID_LOG_RECORD 2 TSA 382
TID_TSRM_QUEUE_RECOVERY_ERR1 2 TSA 382
TID_TSRM_QUEUE_RECOVERY_ERR2 2 TSA 382
TID_TSRM_RECOVERY 2 TSA 382
TID_TSRM_RMDE_INVALID_FUNCTION 2 TSA 382
TID_TSRM_RMKP_INVALID_FUNCTION 2 TSA 382
TID_TSRM_RMRO_INVALID_FUNCTION 2 TSA 382
TID_TSRM_SECTION_RECOVERY_ERR1 2 TSA 382
TID_TSRM_SECTION_RECOVERY_ERR2 2 TSA 382
TID_TSRM_SECTION_RECOVERY_ERR3 2 TSA 382
TID_TSRM_TSIC_INVALID_FUNCTION 2 TSA 382
TID_TSRM_UNLOCK_ERROR_RECOVERY 2 TSA 382
TID_TSSH_AFTER_CLOSE 2 TSA 383
TID_TSSH_AFTER_CONNECT 2 TSA 383
TID_TSSH_AFTER_QUERY_SERVER 2 TSA 383
TID_TSSH_AFTER_SERVER_REQUEST 2 TSA 383
TID_TSSH_BEFORE_CLOSE 2 TSA 383
TID_TSSH_BEFORE_CONNECT 2 TSA 383
TID_TSSH_BEFORE_QUERY_SERVER 2 TSA 383
TID_TSSH_BEFORE_SERVER_REQUEST 2 TSA 383
TID_TSSH_ENTRY 2 TSA 383
TID_TSSH_EXIT 2 TSA 383
TID_TSSH_INVALID_FORMAT 2 TSA 383
TID_TSSH_INVALID_FUNCTION 2 TSA 383
TID_TSSH_RECOVERY 2 TSA 383
TID_TSSH_UNLOCK_ERROR_RECOVERY 2 TSA 383
TID_TSSR_ENTRY 2 TSA 382
TID_TSSR_EXIT 2 TSA 382
TID_TSSR_INVALID_EXIT_POINT 2 TSA 382
TID_TSSR_INVALID_FORMAT 2 TSA 382
TID_TSSR_INVALID_FUNCTION 2 TSA 382
TID_TSSR_RECOVERY 2 TSA 382
TID_TSSR_UNLOCK_ERROR_RECOVERY 2 TSA 382
TID_TSST_ENTRY 2 TSA 382
TID_TSST_EXIT 2 TSA 382
TID_TSST_INVALID_FORMAT 2 TSA 382
TID_TSST_INVALID_FUNCTION 2 TSA 382
TID_TSST_RECOVERY 2 TSA 382
TID_TSST_STATS_BUFFER_TOO_SMALL 2 TSA 382
TID_TSST_UNLOCK_ERROR_RECOVERY 2 TSA 382
TID_TSWQ_AFTER_SUSPEND 2 TSA 383
TID_TSWQ_BEFORE_SUSPEND 2 TSA 383
TID_TSWQ_DSSR_INQUIRE_SUSPEND 2 TSA 383
TID_TSWQ_ENTRY 2 TSA 382
TID_TSWQ_EXIT 2 TSA 382
TID_TSWQ_INVALID_FORMAT 2 TSA 382
TID_TSWQ_INVALID_FUNCTION 2 TSA 382
TID_TSWQ_RECOVERY 2 TSA 382

TID_TSWQ_UNLOCK_ERROR_RECOVERY 2 TSA 382
TID_USAD_ADD_TIMEOUT_FAILED 2 USANC 407
TID_USAD_DCE_EXCEPTION_UNKNOWN 2 USANC 407
TID_USAD_DEL_EXPIRED_FAILED 2 USANC 407
TID_USAD_DEL_TIMEOUT_FAILED 2 USANC 407
TID_USAD_DFHUSER_DEQ_FAILED 2 USANC 407
TID_USAD_ENTRY 2 USANC 407
TID_USAD_EXCEPTION_UNKNOWN 2 USANC 407
TID_USAD_EXIT 2 USANC 407
TID_USAD_EXTRACT_FAILED 2 USANC 407
TID_USAD_INVALID_DCE_STATE 2 USANC 407
TID_USAD_INVALID_FORMAT 2 USANC 407
TID_USAD_INVALID_FUNCTION 2 USANC 407
TID_USAD_INVALID_PARAMETERS 2 USANC 407
TID_USAD_INVALID_SECURITY_TOKEN 2 USANC 407
TID_USAD_LOCK_ERROR 2 USANC 407
TID_USAD_RECOVERY 2 USANC 407
TID_USAD_UDB_PTR_INVALID 2 USANC 407
TID_USAD_UNLOCK_ERROR 2 USANC 407
TID_USAD_UNLOCK_ERROR_RECOVERY 2 USANC 407
TID_USAD_USE_COUNT_ERROR 2 USANC 407
TID_USAD_USER_DIR_ADD_DUPLICATE 2 USANC 407
TID_USAD_USER_DIR_ADD_ERROR 2 USANC 407
TID_USAD_USER_DIR_DELETE_ERROR 2 USANC 407
TID_USAD_USER_NOT_IN_DIRECTORY 2 USANC 407
TID_USDE_DFHUSER_DEQ_FAILED 2 USANC 409
TID_USDE_ENTRY 2 USANC 409
TID_USDE_EXCEPTION_UNKNOWN 2 USANC 409
TID_USDE_EXIT 2 USANC 409
TID_USDE_INVALID_FORMAT 2 USANC 409
TID_USDE_INVALID_FUNCTION 2 USANC 409
TID_USDE_LOCK_ERROR 2 USANC 409
TID_USDE_RECOVERY 2 USANC 409
TID_USDE_UNLOCK_ERROR 2 USANC 409
TID_USDE_UNLOCK_ERROR_RECOVERY 2 USANC 409
TID_USDM_ENTRY 2 USANC 406
TID_USDM_EXIT 2 USANC 406
TID_USDM_GET_PARMS_FAILED 2 USANC 407
TID_USDM_INVALID_FORMAT 2 USANC 406
TID_USDM_INVALID_FUNCTION 2 USANC 406
TID_USDM_NO_STORAGE_FOR_USA 2 USANC 407
TID_USDM_RECOVERY 2 USANC 406
TID_USDM_UNLOCK_ERROR 2 USANC 406
TID_USFL_DEL_TIMEOUT_FAILED 2 USANC 408
TID_USFL_DFHUSER_DEQ_FAILED 2 USANC 408
TID_USFL_ENTRY 2 USANC 408
TID_USFL_EXCEPTION_UNKNOWN 2 USANC 408
TID_USFL_EXIT 2 USANC 408
TID_USFL_INVALID_FORMAT 2 USANC 408
TID_USFL_INVALID_FUNCTION 2 USANC 408
TID_USFL_INVALID_SECURITY_TOKEN 2 USANC 408
TID_USFL_LOCK_ERROR 2 USANC 408
TID_USFL_RECOVERY 2 USANC 408
TID_USFL_UDB_PTR_INVALID 2 USANC 408
TID_USFL_UNFLATTEN_USER_ERROR 2 USANC 408
TID_USFL_UNLOCK_ERROR 2 USANC 408
TID_USFL_UNLOCK_ERROR_RECOVERY 2 USANC 408
TID_USFL_USE_COUNT_ERROR 2 USANC 408
TID_USFL_USER_DIR_ADD_DUPLICATE 2 USANC 408
TID_USFL_USER_DIR_DELETE_ERROR 2 USANC 408
TID_USFL_USER_NOT_IN_DIRECTORY 2 USANC 408
TID_USIS_ENTRY 2 USANC 407
TID_USIS_EXIT 2 USANC 407
TID_USIS_INVALID_FORMAT 2 USANC 407
TID_USIS_INVALID_FUNCTION 2 USANC 407
TID_USIS_LOCK_ERROR 2 USANC 407
TID_USIS_NO_INQUIRE_PARAMETERS 2 USANC 407
TID_USIS_NO_SET_PARAMETERS 2 USANC 407
TID_USIS_RECOVERY 2 USANC 407
TID_USIS_UNLOCK_ERROR 2 USANC 407
TID_USIS_UNLOCK_ERROR_RECOVERY 2 USANC 407
TID_USST_ENTRY 2 USANC 408
TID_USST_EXIT 2 USANC 408
TID_USST_INVALID_FORMAT 2 USANC 408
TID_USST_INVALID_FUNCTION 2 USANC 408
TID_USST_LOCK_ERROR 2 USANC 408
TID_USST_RECOVERY 2 USANC 408
TID_USST_UNLOCK_ERROR 2 USANC 408
TID_USST_UNLOCK_ERROR_RECOVERY 2 USANC 408
TID_USTI_ADD_QUEUE_ENTRY_ERROR 2 USANC 408
TID_USTI_ALREADY_IN_QUEUE 2 USANC 408
TID_USTI_DELETE_QUEUE_ENTRY_ERROR 2 USANC 408
TID_USTI_ENTRY 2 USANC 408
TID_USTI_EXCEPTION_UNKNOWN 2 USANC 408
TID_USTI_EXIT 2 USANC 408
TID_USTI_GET_QUEUE_ENTRY_ERROR 2 USANC 408
TID_USTI_INVALID_FORMAT 2 USANC 408
TID_USTI_INVALID_FUNCTION 2 USANC 408
TID_USTI_LOCK_ERROR 2 USANC 408
TID_USTI_QUEUE_ENTRY_IN_USE 2 USANC 408
TID_USTI_RECOVERY 2 USANC 408
TID_USTI_SET_QUEUE_ENTRY_ERROR 2 USANC 408
TID_USTI_TIMER_CANCEL_REQ_FAILED 2 USANC 409
TID_USTI_TIMER_INTERVAL_REQ_FAILED 2 USANC 408
TID_USTI_UDB_PTR_INVALID 2 USANC 408
TID_USTI_UNLOCK_ERROR 2 USANC 408
TID_USTI_UNLOCK_ERROR_RECOVERY 2 USANC 408
TID_USTI_UTQ_IS_EMPTY 2 USANC 409
TID_USXM_ALREADY_ADDED_SECURITY 2 USANC 408
TID_USXM_BAD_SECURITY_TOKEN 2 USANC 407
TID_USXM_DIRMAN_FAILURE 2 USANC 407
TID_USXM_ENTRY 2 USANC 407
TID_USXM_EXIT 2 USANC 407
TID_USXM_GETMAIN_FAILURE 2 USANC 407
TID_USXM_INVALID_FORMAT 2 USANC 407
TID_USXM_INVALID_FUNCTION 2 USANC 407
TID_USXM_INVALID_TRANSACTION_TOKEN 2 USANC 408
TID_USXM_LOCK_ERROR 2 USANC 407
TID_USXM_NO_PRINCIPAL_UDB_PTR 2 USANC 408
TID_USXM_RECOVERY 2 USANC 407
TID_USXM_TOKEN_TYPE_ERROR 2 USANC 408
TID_USXM_TRAN_USE_COUNT_LOW 2 USANC 407
TID_USXM_TRAN_USE_COUNT_MAX 2 USANC 407
TID_USXM_TRAN_USE_COUNT_NEG 2 USANC 407
TID_USXM_UNLOCK_ERROR 2 USANC 407
TID_USXM_UNLOCK_ERROR_RECOVERY 2 USANC 407
TID_USXM_USAD_ERROR 2 USANC 408
TID_XSAD_ENTRY 2 XSANC 449
TID_XSAD_EXIT 2 XSANC 449
TID_XSAD_INVALID_FORMAT 2 XSANC 449
TID_XSAD_INVALID_FUNCTION 2 XSANC 449
TID_XSAD_RECOVERY 2 XSANC 449
TID_XSAD_XSSA_FAILURE 2 XSANC 449
TID_XSAD_XSSB_FAILURE 2 XSANC 449
TID_XSDM_ENTRY 2 XSANC 449
TID_XSDM_EXIT 2 XSANC 449
TID_XSDM_GET_PARMS_FAILED 2 XSANC 449
TID_XSDM_GET_SVC_ERROR 2 XSANC 449
TID_XSDM_INVALID_FORMAT 2 XSANC 449
TID_XSDM_INVALID_FUNCTION 2 XSANC 449
TID_XSDM_LOCK_ERROR 2 XSANC 449
TID_XSDM_NO_STORAGE_FOR_XSA 2 XSANC 449
TID_XSDM_RECOVERY 2 XSANC 449
TID_XSDM_UNLOCK_ERROR 2 XSANC 449
TID_XSFL_DISASTROUS_ERROR_IN_XSSA 2 XSANC 450
TID_XSFL_ENTRY 2 XSANC 450
TID_XSFL_EXIT 2 XSANC 450
TID_XSFL_INVALID_FLATTENED_BUFFER 2 XSANC 450
TID_XSFL_INVALID_FORMAT 2 XSANC 450
TID_XSFL_INVALID_FORMAT_PASSED_TO_XSSA 2 XSANC 450
TID_XSFL_INVALID_FUNCTION 2 XSANC 450
TID_XSFL_INVALID_FUNCTION_PASSED_TO_XSSA 2 XSANC 450
TID_XSFL_INVALID_SECURITY_TOKEN 2 XSANC 450
TID_XSFL_RECOVERY 2 XSANC 450
TID_XSIS_ENTRY 2 XSANC 449
TID_XSIS_EXIT 2 XSANC 449
TID_XSIS_EXTRACT_LOCK_ERROR 2 XSANC 449
TID_XSIS_EXTRACT_UNLOCK_ERROR 2 XSANC 449
TID_XSIS_INVALID_FORMAT 2 XSANC 449
TID_XSIS_INVALID_FUNCTION 2 XSANC 449
TID_XSIS_REBUILD_LOCK_ERROR 2 XSANC 449
TID_XSIS_REBUILD_UNLOCK_ERROR 2 XSANC 449
TID_XSIS_RECOVERY 2 XSANC 449
TID_XSIS_XSSC_FAILURE 2 XSANC 449
TID_XSIS_XSSI_FAILURE 2 XSANC 449
TID_XSLU_ENTRY 2 XSANC 450
TID_XSLU_ESTAE_FAILURE 2 XSANC 450
TID_XSLU_EXIT 2 XSANC 450
TID_XSLU_EXTRACT_FAILURE 2 XSANC 450
TID_XSLU_EXTRACT_LOCK_ERROR 2 XSANC 450
TID_XSLU_EXTRACT_UNLOCK_ERROR 2 XSANC 450
TID_XSLU_INVALID_FORMAT 2 XSANC 450
TID_XSLU_INVALID_FUNCTION 2 XSANC 450
TID_XSLU_RECOVERY 2 XSANC 450
TID_XSLU_XSSB_FAILURE 2 XSANC 450
TID_XSPW_ENTRY 2 XSANC 450

TID_XSPW_EXIT 2 XSANC 450
TID_XSPW_INVALID_FORMAT 2 XSANC 450
TID_XSPW_INVALID_FUNCTION 2 XSANC 450
TID_XSPW_RECOVERY 2 XSANC 450
TID_XSPW_XSSB_FAILURE 2 XSANC 450
TID_XSPW_XSSD_FAILURE 2 XSANC 450
TID_XSPW_XSSE_FAILURE 2 XSANC 450
TID_XSRC_DISPATCHER_ERROR 2 XSANC 450
TID_XSRC_ENTRY 2 XSANC 450
TID_XSRC_EXIT 2 XSANC 450
TID_XSRC_INVALID_ACCESS 2 XSANC 450
TID_XSRC_INVALID_FORMAT 2 XSANC 450
TID_XSRC_INVALID_FUNCTION 2 XSANC 450
TID_XSRC_INVALID_RESOURCE_TYPE 2 XSANC 450
TID_XSRC_LOCK_ERROR 2 XSANC 450
TID_XSRC_RECOVERY 2 XSANC 450
TID_XSRC_RESOURCE_CHECK_ENTRY 2 XSANC 450
TID_XSRC_RESOURCE_CHECK_ERROR 2 XSANC 450
TID_XSRC_RESOURCE_CHECK_EXIT 2 XSANC 450
TID_XSRC_UNLOCK_ERROR 2 XSANC 450
TID_XSRC_XRF_TRACKING_ERROR 2 XSANC 450
TID_XSRC_XSSC_FAILURE 2 XSANC 450
TID_XSS_ENTRY 2 XSANC 450
TID_XSS_EXCEPTION 2 XSANC 450
TID_XSS_EXIT 2 XSANC 450
TID_XSS_INSTALLATION_DATA 2 XSANC 450
TID_XSS_SVC_ERROR 2 XSANC 450
TID_XSXM_ENTRY 2 XSANC 449
TID_XSXM_EXIT 2 XSANC 449
TID_XSXM_GETMAIN_FAILURE 2 XSANC 450
TID_XSXM_INVALID_FORMAT 2 XSANC 449
TID_XSXM_INVALID_FUNCTION 2 XSANC 450
TID_XSXM_RECOVERY 2 XSANC 449
TIDM_NAME 7 TIA 380
TIME (30) L2BL 208
TIME_OF_LAST_MOVE 223
TIME_OUT_GAP (48) DSANC 54
TIME_PERIOD (BIT) STUCB 376
TIME_PERIOD_SELECTED (BIT) STUCB 376
TIMEOUT_ACTIVE (BIT) RMLK 312
TIMEOUT_ACTIVE (BIT) RMUW 331
TIMEOUT_POINTER (34) SOA 371
TIMEOUT_TIME (38) DSTSK 65
TIMEOUT_TYPE (76) DSTSK 66
timer
    timer domain anchor block, TIA 378
TIMER (120) DSANC 56
TIMER_REQUEST_ELEMENT (0) TIA 378
TIMER_TOKEN (540) RMLK 315
TIMER_TOKEN (540) RMUW 334
TIMES_LOGGED (5C) RMLK 310
TIMES_LOGGED (8) RMLK 317
TIMES_LOGGED (964) RMLK 308
TIMES_RESTORED 308, 311
TIQC_SUBPOOL_TOKEN (28) TIA 378
TISR_NAME 7 TIA 380
TMA_ARROW (2) MNCBS 262
TMA_BEGIN (D8) MNCBS 263
TMA_BLOCK_ID (8) MNCBS 262
TMA_CELL_POOL_NAME 8 MNCBS 273
TMA_CHILD_TMA (24) MNCBS 262
TMA_CLASS_STATUS (44) MNCBS 262
TMA_CLOCKS (440) MNCBS 265
TMA_COMPOSITE_171_INTVL (78) MNCBS 263
TMA_COMPOSITE_171_INTVL_COUNT (7C) MNCBS 263
TMA_COMPOSITE_254_INTVL (80) MNCBS 263
TMA_COMPOSITE_254_INTVL_COUNT (84) MNCBS 263
TMA_CPU_TIME (58) MNCBS 262
TMA_CREATION_STCK (10) MNCBS 262
TMA_CURRENT 263
TMA_DEPTH_COUNT (28) MNCBS 262
TMA_DFH (3) MNCBS 262
TMA_DFHCBTS_200 (180) MNCBS 263
TMA_DFHCBTS_201 (1A4) MNCBS 263
TMA_DFHCBTS_202 (1AC) MNCBS 263
TMA_DFHCBTS_203 (1E0) MNCBS 263
TMA_DFHCBTS_204 (214) MNCBS 263
TMA_DFHCBTS_205 (3AC) MNCBS 264
TMA_DFHCBTS_206 (3B0) MNCBS 264
TMA_DFHCBTS_207 (3B4) MNCBS 264
TMA_DFHCBTS_208 (3B8) MNCBS 264
TMA_DFHCBTS_209 (3BC) MNCBS 264
TMA_DFHCBTS_210 (3C0) MNCBS 264
TMA_DFHCBTS_211 (3C4) MNCBS 265
TMA_DFHCBTS_212 (3C8) MNCBS 265
TMA_DFHCBTS_213 (3CC) MNCBS 265
TMA_DFHCBTS_214 (3D0) MNCBS 265
TMA_DFHCBTS_215 (3D4) MNCBS 265
TMA_DFHCBTS_216 (3D8) MNCBS 265
TMA_DFHCBTS_217 (3DC) MNCBS 265
TMA_DFHCBTS_218 (3E0) MNCBS 265
TMA_DFHCBTS_219 (3E4) MNCBS 265
TMA_DFHCBTS_220 (3E8) MNCBS 265
TMA_DFHCBTS_221 (3EC) MNCBS 265
TMA_DFHCBTS_222 (3F0) MNCBS 265
TMA_DFHCICS_005 (EC) MNCBS 263
TMA_DFHCICS_006 (F4) MNCBS 263
TMA_DFHCICS_025 (384) MNCBS 264
TMA_DFHCICS_089 (E0) MNCBS 263
TMA_DFHCICS_103 (4A8) MNCBS 266
TMA_DFHCICS_103_COUNT (4AD) MNCBS 266
TMA_DFHCICS_103_FLAG (4AC) MNCBS 266
TMA_DFHCICS_103_TIME (4A8) MNCBS 266
TMA_DFHCICS_112 (25C) MNCBS 263
TMA_DFHCICS_130 (138) MNCBS 263
TMA_DFHCICS_131 (13C) MNCBS 263
TMA_DFHCICS_167 (148) MNCBS 263
TMA_DFHCICS_168 (150) MNCBS 263
TMA_DFHDATA_179 (430) MNCBS 265
TMA_DFHDATA_180 (434) MNCBS 265
TMA_DFHDATA_186 (5B0) MNCBS 269
TMA_DFHDATA_186_COUNT (5B5) MNCBS 269
TMA_DFHDATA_186_FLAG (5B4) MNCBS 269
TMA_DFHDATA_186_TIME (5B0) MNCBS 269
TMA_DFHDATA_187 (5B8) MNCBS 269
TMA_DFHDATA_187_COUNT (5BD) MNCBS 269
TMA_DFHDATA_187_FLAG (5BC) MNCBS 269
TMA_DFHDATA_187_TIME (5B8) MNCBS 269
TMA_DFHDATA_188 (5C0) MNCBS 269
TMA_DFHDATA_188_COUNT (5C5) MNCBS 269
TMA_DFHDATA_188_FLAG (5C4) MNCBS 269
TMA_DFHDATA_188_TIME (5C0) MNCBS 269
TMA_DFHDATA_189 (5C8) MNCBS 269
TMA_DFHDATA_189_COUNT (5CD) MNCBS 269
TMA_DFHDATA_189_FLAG (5CC) MNCBS 269
TMA_DFHDATA_189_TIME (5C8) MNCBS 269
TMA_DFHDEST_041 (32C) MNCBS 264
TMA_DFHDEST_042 (330) MNCBS 264
TMA_DFHDEST_043 (334) MNCBS 264
TMA_DFHDEST_091 (338) MNCBS 264
TMA_DFHDEST_101 (4D8) MNCBS 266
TMA_DFHDEST_101_COUNT (4DD) MNCBS 266
TMA_DFHDEST_101_FLAG (4DC) MNCBS 266
TMA_DFHDEST_101_TIME (4D8) MNCBS 266
TMA_DFHDOCH_226 (410) MNCBS 265
TMA_DFHDOCH_227 (414) MNCBS 265
TMA_DFHDOCH_228 (418) MNCBS 265
TMA_DFHDOCH_229 (41C) MNCBS 265
TMA_DFHDOCH_230 (420) MNCBS 265
TMA_DFHDOCH_240 (424) MNCBS 265
TMA_DFHFEPI_150 (388) MNCBS 264
TMA_DFHFEPI_151 (38C) MNCBS 264
TMA_DFHFEPI_152 (390) MNCBS 264
TMA_DFHFEPI_153 (394) MNCBS 264
TMA_DFHFEPI_154 (398) MNCBS 264
TMA_DFHFEPI_155 (39C) MNCBS 264
TMA_DFHFEPI_156 (520) MNCBS 267
TMA_DFHFEPI_156_COUNT (525) MNCBS 267
TMA_DFHFEPI_156_FLAG (524) MNCBS 267
TMA_DFHFEPI_156_TIME (520) MNCBS 267
TMA_DFHFEPI_157 (3A0) MNCBS 264
TMA_DFHFEPI_158 (3A4) MNCBS 264
TMA_DFHFEPI_159 (3A8) MNCBS 264
TMA_DFHFILE_036 (310) MNCBS 264
TMA_DFHFILE_037 (314) MNCBS 264
TMA_DFHFILE_038 (318) MNCBS 264
TMA_DFHFILE_039 (31C) MNCBS 264
TMA_DFHFILE_040 (320) MNCBS 264
TMA_DFHFILE_063 (4B8) MNCBS 266
TMA_DFHFILE_063_COUNT (4BD) MNCBS 266
TMA_DFHFILE_063_FLAG (4BC) MNCBS 266
TMA_DFHFILE_063_TIME (4B8) MNCBS 266
TMA_DFHFILE_070 (328) MNCBS 264
TMA_DFHFILE_093 (324) MNCBS 264
TMA_DFHFILE_174 (540) MNCBS 267
TMA_DFHFILE_174_COUNT (545) MNCBS 268

TMA_DFHFILE_174_FLAG (544) MNCBS 267
TMA_DFHFILE_174_TIME (540) MNCBS 267
TMA_DFHFILE_175 (548) MNCBS 268
TMA_DFHFILE_175_COUNT (54D) MNCBS 268
TMA_DFHFILE_175_FLAG (54C) MNCBS 268
TMA_DFHFILE_175_TIME (548) MNCBS 268
TMA_DFHFILE_176 (580) MNCBS 268
TMA_DFHFILE_176_COUNT (585) MNCBS 268
TMA_DFHFILE_176_FLAG (584) MNCBS 268
TMA_DFHFILE_176_TIME (580) MNCBS 268
TMA_DFHJOUR_010 (4C0) MNCBS 266
TMA_DFHJOUR_010_COUNT (4C5) MNCBS 266
TMA_DFHJOUR_010_FLAG (4C4) MNCBS 266
TMA_DFHJOUR_010_TIME (4C0) MNCBS 266
TMA_DFHJOUR_058 (370) MNCBS 264
TMA_DFHJOUR_172 (374) MNCBS 264
TMA_DFHMAPP_050 (34C) MNCBS 264
TMA_DFHMAPP_051 (350) MNCBS 264
TMA_DFHMAPP_052 (354) MNCBS 264
TMA_DFHMAPP_090 (358) MNCBS 264
TMA_DFHPROG_055 (35C) MNCBS 264
TMA_DFHPROG_056 (360) MNCBS 264
TMA_DFHPROG_057 (364) MNCBS 264
TMA_DFHPROG_071 (114) MNCBS 263
TMA_DFHPROG_072 (368) MNCBS 264
TMA_DFHPROG_073 (36C) MNCBS 264
TMA_DFHPROG_113 (254) MNCBS 263
TMA_DFHPROG_114 (258) MNCBS 263
TMA_DFHPROG_115 (4E0) MNCBS 266
TMA_DFHPROG_115_COUNT (4E5) MNCBS 267
TMA_DFHPROG_115_FLAG (4E4) MNCBS 267
TMA_DFHPROG_115_TIME (4E0) MNCBS 267
TMA_DFHSOCK_241 (5A8) MNCBS 268
TMA_DFHSOCK_241_COUNT (5AD) MNCBS 269
TMA_DFHSOCK_241_FLAG (5AC) MNCBS 269
TMA_DFHSOCK_241_TIME (5A8) MNCBS 269
TMA_DFHSOCK_242 (428) MNCBS 265
TMA_DFHSOCK_243 (42C) MNCBS 265
TMA_DFHSOCK_244 (224) MNCBS 263
TMA_DFHSTOR_033 (2A4) MNCBS 264
TMA_DFHSTOR_033_C (90) MNCBS 263
TMA_DFHSTOR_054 (294) MNCBS 264
TMA_DFHSTOR_087 (2EC) MNCBS 264
TMA_DFHSTOR_087_C (A0) MNCBS 263
TMA_DFHSTOR_095 (2B4) MNCBS 264
TMA_DFHSTOR_095_O (C4) MNCBS 263
TMA_DFHSTOR_105 (298) MNCBS 264
TMA_DFHSTOR_106 (2A8) MNCBS 264
TMA_DFHSTOR_106_C (94) MNCBS 263
TMA_DFHSTOR_107 (2BC) MNCBS 264
TMA_DFHSTOR_107_O (C8) MNCBS 263
TMA_DFHSTOR_108 (2F4) MNCBS 264
TMA_DFHSTOR_108_C (A8) MNCBS 263
TMA_DFHSTOR_116 (2AC) MNCBS 264
TMA_DFHSTOR_116_C (98) MNCBS 263
TMA_DFHSTOR_117 (29C) MNCBS 264
TMA_DFHSTOR_118 (2C4) MNCBS 264
TMA_DFHSTOR_118_O (CC) MNCBS 263
TMA_DFHSTOR_119 (2B0) MNCBS 264
TMA_DFHSTOR_119_C (9C) MNCBS 263
TMA_DFHSTOR_120 (2A0) MNCBS 264
TMA_DFHSTOR_121 (2CC) MNCBS 264
TMA_DFHSTOR_121_O (D0) MNCBS 263
TMA_DFHSTOR_122 (300) MNCBS 264
TMA_DFHSTOR_122_C (B4) MNCBS 263
TMA_DFHSTOR_139 (2F0) MNCBS 264
TMA_DFHSTOR_139_C (A4) MNCBS 263
TMA_DFHSTOR_142 (2F8) MNCBS 264
TMA_DFHSTOR_142_C (AC) MNCBS 263
TMA_DFHSTOR_143 (2FC) MNCBS 264
TMA_DFHSTOR_143_C (B0) MNCBS 263
TMA_DFHSTOR_144 (2D4) MNCBS 264
TMA_DFHSTOR_145 (2D8) MNCBS 264
TMA_DFHSTOR_146 (2DC) MNCBS 264
TMA_DFHSTOR_147 (2E0) MNCBS 264
TMA_DFHSTOR_148 (2E4) MNCBS 264
TMA_DFHSTOR_149 (2E8) MNCBS 264
TMA_DFHSTOR_160 (30C) MNCBS 264
TMA_DFHSTOR_160_C (C0) MNCBS 263
TMA_DFHSTOR_161 (308) MNCBS 264
TMA_DFHSTOR_161_C (BC) MNCBS 263
TMA_DFHSTOR_162 (304) MNCBS 264
TMA_DFHSTOR_162_C (B8) MNCBS 263
TMA_DFHSYNC_060 (380) MNCBS 264
TMA_DFHSYNC_173 (538) MNCBS 267
TMA_DFHSYNC_173_COUNT (53D) MNCBS 267
TMA_DFHSYNC_173_FLAG (53C) MNCBS 267
TMA_DFHSYNC_173_TIME (538) MNCBS 267
TMA_DFHSYNC_177 (588) MNCBS 268
TMA_DFHSYNC_177_COUNT (58D) MNCBS 268
TMA_DFHSYNC_177_FLAG (58C) MNCBS 268
TMA_DFHSYNC_177_TIME (588) MNCBS 268
TMA_DFHSYNC_196 (5A0) MNCBS 268
TMA_DFHSYNC_196_COUNT (5A5) MNCBS 268
TMA_DFHSYNC_196_FLAG (5A4) MNCBS 268
TMA_DFHSYNC_196_TIME (5A0) MNCBS 268
TMA_DFHTASK_001 (D8) MNCBS 263
TMA_DFHTASK_004 (E8) MNCBS 263
TMA_DFHTASK_007 (440) MNCBS 265
TMA_DFHTASK_007_COUNT (445) MNCBS 265
TMA_DFHTASK_007_FLAG (444) MNCBS 265
TMA_DFHTASK_007_TIME (440) MNCBS 265
TMA_DFHTASK_008 (448) MNCBS 265
TMA_DFHTASK_008_COUNT (44D) MNCBS 265
TMA_DFHTASK_008_FLAG (44C) MNCBS 265
TMA_DFHTASK_008_TIME (448) MNCBS 265
TMA_DFHTASK_014 (450) MNCBS 265
TMA_DFHTASK_014_COUNT (455) MNCBS 265
TMA_DFHTASK_014_FLAG (454) MNCBS 265
TMA_DFHTASK_014_TIME (450) MNCBS 265
TMA_DFHTASK_031 (FC) MNCBS 263
TMA_DFHTASK_059 (378) MNCBS 264
TMA_DFHTASK_064 (250) MNCBS 263
TMA_DFHTASK_066 (37C) MNCBS 264
TMA_DFHTASK_082 (234) MNCBS 263
TMA_DFHTASK_097 (11C) MNCBS 263
TMA_DFHTASK_098 (130) MNCBS 263
TMA_DFHTASK_102 (458) MNCBS 265
TMA_DFHTASK_102_COUNT (45D) MNCBS 265
TMA_DFHTASK_102_FLAG (45C) MNCBS 265
TMA_DFHTASK_102_TIME (458) MNCBS 265
TMA_DFHTASK_109 (100) MNCBS 263
TMA_DFHTASK_123 (508) MNCBS 267
TMA_DFHTASK_123_COUNT (50D) MNCBS 267
TMA_DFHTASK_123_FLAG (50C) MNCBS 267
TMA_DFHTASK_123_TIME (508) MNCBS 267
TMA_DFHTASK_124 (16C) MNCBS 263
TMA_DFHTASK_125 (4E8) MNCBS 267
TMA_DFHTASK_125_COUNT (4ED) MNCBS 267
TMA_DFHTASK_125_FLAG (4EC) MNCBS 267
TMA_DFHTASK_125_TIME (4E8) MNCBS 267
TMA_DFHTASK_126 (4F0) MNCBS 267
TMA_DFHTASK_126_COUNT (4F5) MNCBS 267
TMA_DFHTASK_126_FLAG (4F4) MNCBS 267
TMA_DFHTASK_126_TIME (4F0) MNCBS 267
TMA_DFHTASK_127 (4F8) MNCBS 267
TMA_DFHTASK_127_COUNT (4FD) MNCBS 267
TMA_DFHTASK_127_FLAG (4FC) MNCBS 267
TMA_DFHTASK_127_TIME (4F8) MNCBS 267
TMA_DFHTASK_128 (550) MNCBS 268
TMA_DFHTASK_128_COUNT (555) MNCBS 268
TMA_DFHTASK_128_FLAG (554) MNCBS 268
TMA_DFHTASK_128_TIME (550) MNCBS 268
TMA_DFHTASK_129 (500) MNCBS 267
TMA_DFHTASK_129_COUNT (505) MNCBS 267
TMA_DFHTASK_129_FLAG (504) MNCBS 267
TMA_DFHTASK_129_TIME (500) MNCBS 267
TMA_DFHTASK_132 (140) MNCBS 263
TMA_DFHTASK_163 (158) MNCBS 263
TMA_DFHTASK_164 (15C) MNCBS 263
TMA_DFHTASK_166 (104) MNCBS 263
TMA_DFHTASK_170 (528) MNCBS 267
TMA_DFHTASK_170_A (D4) MNCBS 263
TMA_DFHTASK_170_COUNT (52D) MNCBS 267
TMA_DFHTASK_170_FLAG (52C) MNCBS 267
TMA_DFHTASK_170_TIME (528) MNCBS 267
TMA_DFHTASK_171 (530) MNCBS 267
TMA_DFHTASK_171_COUNT (535) MNCBS 267
TMA_DFHTASK_171_FLAG (534) MNCBS 267
TMA_DFHTASK_171_TIME (530) MNCBS 267
TMA_DFHTASK_181 (558) MNCBS 268
TMA_DFHTASK_181_COUNT (55D) MNCBS 268
TMA_DFHTASK_181_FLAG (55C) MNCBS 268
TMA_DFHTASK_181_TIME (558) MNCBS 268
TMA_DFHTASK_182 (560) MNCBS 268
TMA_DFHTASK_182_COUNT (565) MNCBS 268

TMA_DFHTASK_182_FLAG (564) MNCBS 268
TMA_DFHTASK_182_TIME (560) MNCBS 268
TMA_DFHTASK_183 (568) MNCBS 268
TMA_DFHTASK_183_COUNT (56D) MNCBS 268
TMA_DFHTASK_183_FLAG (56C) MNCBS 268
TMA_DFHTASK_183_TIME (568) MNCBS 268
TMA_DFHTASK_184 (570) MNCBS 268
TMA_DFHTASK_184_COUNT (575) MNCBS 268
TMA_DFHTASK_184_FLAG (574) MNCBS 268
TMA_DFHTASK_184_TIME (570) MNCBS 268
TMA_DFHTASK_190 (170) MNCBS 263
TMA_DFHTASK_191 (590) MNCBS 268
TMA_DFHTASK_191_COUNT (595) MNCBS 268
TMA_DFHTASK_191_FLAG (594) MNCBS 268
TMA_DFHTASK_191_TIME (590) MNCBS 268
TMA_DFHTASK_195 (598) MNCBS 268
TMA_DFHTASK_195_COUNT (59D) MNCBS 268
TMA_DFHTASK_195_FLAG (59C) MNCBS 268
TMA_DFHTASK_195_TIME (598) MNCBS 268
TMA_DFHTASK_248 (438) MNCBS 265
TMA_DFHTASK_249 (498) MNCBS 266
TMA_DFHTASK_249_COUNT (49D) MNCBS 266
TMA_DFHTASK_249_FLAG (49C) MNCBS 266
TMA_DFHTASK_249_TIME (498) MNCBS 266
TMA_DFHTASK_250 (4A0) MNCBS 266
TMA_DFHTASK_250_COUNT (4A5) MNCBS 266
TMA_DFHTASK_250_FLAG (4A4) MNCBS 266
TMA_DFHTASK_250_TIME (4A0) MNCBS 266
TMA_DFHTASK_251 (43C) MNCBS 265
TMA_DFHTASK_253 (5D0) MNCBS 269
TMA_DFHTASK_253_COUNT (5D5) MNCBS 269
TMA_DFHTASK_253_FLAG (5D4) MNCBS 269
TMA_DFHTASK_253_TIME (5D0) MNCBS 269
TMA_DFHTASK_254 (5D8) MNCBS 269
TMA_DFHTASK_254_COUNT (5DD) MNCBS 269
TMA_DFHTASK_254_FLAG (5DC) MNCBS 269
TMA_DFHTASK_254_TIME (5D8) MNCBS 269
TMA_DFHTASK_255 (460) MNCBS 265
TMA_DFHTASK_255_COUNT (465) MNCBS 265
TMA_DFHTASK_255_FLAG (464) MNCBS 265
TMA_DFHTASK_255_TIME (460) MNCBS 265
TMA_DFHTASK_256 (468) MNCBS 265
TMA_DFHTASK_256_COUNT (46D) MNCBS 265
TMA_DFHTASK_256_FLAG (46C) MNCBS 265
TMA_DFHTASK_256_TIME (468) MNCBS 265
TMA_DFHTASK_257 (470) MNCBS 265
TMA_DFHTASK_257_COUNT (475) MNCBS 265
TMA_DFHTASK_257_FLAG (474) MNCBS 265
TMA_DFHTASK_257_TIME (470) MNCBS 265
TMA_DFHTASK_258 (478) MNCBS 265
TMA_DFHTASK_258_COUNT (47D) MNCBS 266
TMA_DFHTASK_258_FLAG (47C) MNCBS 265
TMA_DFHTASK_258_TIME (478) MNCBS 265
TMA_DFHTASK_259 (480) MNCBS 266
TMA_DFHTASK_259_COUNT (485) MNCBS 266
TMA_DFHTASK_259_FLAG (484) MNCBS 266
TMA_DFHTASK_259_TIME (480) MNCBS 266
TMA_DFHTASK_260 (488) MNCBS 266
TMA_DFHTASK_260_COUNT (48D) MNCBS 266
TMA_DFHTASK_260_FLAG (48C) MNCBS 266
TMA_DFHTASK_260_TIME (488) MNCBS 266
TMA_DFHTASK_261 (490) MNCBS 266
TMA_DFHTASK_261_COUNT (495) MNCBS 266
TMA_DFHTASK_261_FLAG (494) MNCBS 266
TMA_DFHTASK_261_TIME (490) MNCBS 266
TMA_DFHTEMP_011 (4C8) MNCBS 266
TMA_DFHTEMP_011_COUNT (4CD) MNCBS 266
TMA_DFHTEMP_011_FLAG (4CC) MNCBS 266
TMA_DFHTEMP_011_TIME (4C8) MNCBS 266
TMA_DFHTEMP_044 (33C) MNCBS 264
TMA_DFHTEMP_046 (340) MNCBS 264
TMA_DFHTEMP_047 (344) MNCBS 264
TMA_DFHTEMP_092 (348) MNCBS 264
TMA_DFHTEMP_178 (578) MNCBS 268
TMA_DFHTEMP_178_COUNT (57D) MNCBS 268
TMA_DFHTEMP_178_FLAG (57C) MNCBS 268
TMA_DFHTEMP_178_TIME (578) MNCBS 268
TMA_DFHTERM_002 (DC) MNCBS 263
TMA_DFHTERM_009 (4B0) MNCBS 266
TMA_DFHTERM_009_COUNT (4B5) MNCBS 266
TMA_DFHTERM_009_FLAG (4B4) MNCBS 266
TMA_DFHTERM_009_TIME (4B0) MNCBS 266
TMA_DFHTERM_034 (260) MNCBS 263
TMA_DFHTERM_035 (268) MNCBS 263
TMA_DFHTERM_067 (270) MNCBS 263
TMA_DFHTERM_068 (278) MNCBS 263
TMA_DFHTERM_069 (290) MNCBS 264
TMA_DFHTERM_083 (264) MNCBS 263
TMA_DFHTERM_084 (26C) MNCBS 263
TMA_DFHTERM_085 (274) MNCBS 263
TMA_DFHTERM_086 (27C) MNCBS 264
TMA_DFHTERM_100 (4D0) MNCBS 266
TMA_DFHTERM_100_COUNT (4D5) MNCBS 266
TMA_DFHTERM_100_FLAG (4D4) MNCBS 266
TMA_DFHTERM_100_TIME (4D0) MNCBS 266
TMA_DFHTERM_111 (10C) MNCBS 263
TMA_DFHTERM_133 (510) MNCBS 267
TMA_DFHTERM_133_COUNT (515) MNCBS 267
TMA_DFHTERM_133_FLAG (514) MNCBS 267
TMA_DFHTERM_133_TIME (510) MNCBS 267
TMA_DFHTERM_134 (518) MNCBS 267
TMA_DFHTERM_134_COUNT (51D) MNCBS 267
TMA_DFHTERM_134_FLAG (51C) MNCBS 267
TMA_DFHTERM_134_TIME (518) MNCBS 267
TMA_DFHTERM_135 (280) MNCBS 264
TMA_DFHTERM_136 (288) MNCBS 264
TMA_DFHTERM_137 (284) MNCBS 264
TMA_DFHTERM_138 (28C) MNCBS 264
TMA_DFHTERM_165 (164) MNCBS 263
TMA_DFHTERM_169 (168) MNCBS 263
TMA_DFHWEBB_231 (3F4) MNCBS 265
TMA_DFHWEBB_232 (3F8) MNCBS 265
TMA_DFHWEBB_233 (3FC) MNCBS 265
TMA_DFHWEBB_234 (400) MNCBS 265
TMA_DFHWEBB_235 (404) MNCBS 265
TMA_DFHWEBB_236 (408) MNCBS 265
TMA_DFHWEBB_237 (40C) MNCBS 265
TMA_DOMAIN (6) MNCBS 262
TMA_DS_TOKEN (34) MNCBS 262
TMA_ELAPSED_TIME 262
TMA_EXCEPTION_COUNT (48) MNCBS 262
TMA_EXCEPTION_STATUS (BIT) MNCBS 262
TMA_ID_STRING 8 MNCBS 273
TMA_LAST_SUSPEND_INTERVAL (70) MNCBS 263
TMA_LENGTH (0) MNCBS 262
TMA_MNA_PTR (40) MNCBS 262
TMA_OCCUPANCY (C4) MNCBS 263
TMA_PARENT_TMA (20) MNCBS 262
TMA_PERFORMANCE_STATUS (BIT) MNCBS 262
TMA_PREFIX (0) MNCBS 262
TMA_RECURSE_COUNTS (D4) MNCBS 263
TMA_RESERVED_1 (18) MNCBS 262
TMA_RESERVED_2 (2C) MNCBS 262
TMA_RESERVED_3 (3C) MNCBS 262
TMA_RESET (250) MNCBS 263
TMA_START_TIME 262
TMA_SYSEVENT_STATUS (BIT) MNCBS 262
TMA_USER_AREA (5E0) MNCBS 269
TMA_USER_AREA_PTR (30) MNCBS 262
TMA_WLM_SRC_TOKEN (38) MNCBS 262
TO_BE_CLEAR_PENDED (BIT) RMLK 308, 310
token
  log manager record token class, L2RT 239
  security domain transaction token, XSXT 456
  user domain transaction token, USXT 411
TOKEN (2BC) LDCBS 171
TOKEN (60) LDCBS 165
TOKEN2 (40) LDCBS 170
TONR_PTR (C) RDAB 299
TOTAL_HEURISTIC_MISMATCHES (904) RMLK 307
TOTAL_REC_LENGTH (8B0) STUCB 376
TOTAL_REC_PTR (8AC) STUCB 376
TOTAL_RESYNCS (900) RMLK 307
TOTAL_SHUNTED_INDOUBT (978) RMUW 340
TOTAL_SHUNTED_RO_FAIL (97C) RMUW 340
TOTAL_SYNC_BWDS (96C) RMUW 339
TOTAL_SYNC_FWDS (968) RMUW 339
TOTAL_TIME_SHUNTED_INDOUBT (970) RMUW 339
TOTAL_TIME_SHUNTED_RO_FAIL (980) RMUW 340
TP_NAME (1A) PTE 299
TP_NAME (32) PTE 298
TP_NAME (78) CPCPS 33
TP_NAME_LENGTH (18) PTE 299
TP_NAME_LENGTH (30) PTE 298
TP_NAME_LENGTH (74) CPCPS 33
TPE (0) SMMCC 365

TPE_CLASS (0) SMMCC 365
TPE_INITIMG (1) SMMCC 365
TPE_LENGTH (2) SMMCC 365
TPE_LIOA_DATA_START (8) SMMCC 365
TPE_NEXT (4) SMMCC 365
TPE_SAA (0) SMMCC 365
TPE_TIOA_DATA_START (D) SMMCC 365
TPE_TIOA_PREFIX (8) SMMCC 365
TPID_PADM_ENTRY 2 PAA 284
TPID_PADM_EXIT 2 PAA 284
TPID_PADM_INV_FORMAT 2 PAA 284
TPID_PADM_INV_FUNCTION 2 PAA 284
TPID_PADM_RECOVERY 2 PAA 284
TPID_PAGP_AWTOR 2 PAA 284
TPID_PAGP_BWTOR 2 PAA 284
TPID_PAGP_ENTRY 2 PAA 284
TPID_PAGP_EXIT 2 PAA 284
TPID_PAGP_INV_FORMAT 2 PAA 284
TPID_PAGP_INV_FUNCTION 2 PAA 284
TPID_PAGP_INVDC 2 PAA 284
TPID_PAGP_INVRQDOM 2 PAA 284
TPID_PAGP_INVSIT 2 PAA 284
TPID_PAGP_RECOVERY 2 PAA 284
TPID_PASY_ENTRY 2 PAA 284
TPID_PASY_EXIT 2 PAA 284
TPID_TIDM_ENTRY 2 TIA 379
TPID_TIDM_EXIT 2 TIA 379
TPID_TIDM_INVDC 2 TIA 379
TPID_TIDM_INVFMT 2 TIA 379
TPID_TIDM_RECOV 2 TIA 379
TPID_TISR_BADSTCK 2 TIA 379
TPID_TISR_ENTRY 2 TIA 379
TPID_TISR_EXIT 2 TIA 379
TPID_TISR_INVDC 2 TIA 379
TPID_TISR_INVFMT 2 TIA 379
TPID_TISR_NOATTACH 2 TIA 379
TPID_TISR_RECOV 2 TIA 379
TPID_TISR_TOOLATE 2 TIA 379
TPID_TISR_XINTVL 2 TIA 379
TPID_TISR_XTOKEN 2 TIA 379
trace
    CICS affinities utility trace table, CAUTR 26
    frontend programming interface trace, FEP01 108
TRACE_TABLE 27
tracker
    log manager lock tracker class, L2LT 238
TRAN_CONTEXT (0) RMUW 336
TRAN_CONTEXT (33) RMLK 311
TRAN_CONTEXT (33) RMUW 331
TRAN_TOKEN (14) RMUW 336
TRAN_TOKEN (47) RMLK 311
TRAN_TOKEN (47) RMUW 331
tran.
    transaction manager tran. browse element, XMXBC 441
TRANDEF_CATALOG_RECORD (0) XMCAT 438
TRANID 11, 13, 18
TRANID (10) RMUW 336
TRANID (43) RMLK 311
TRANID (43) RMUW 331
TRANISO (BIT) DSANC 58
TRANNUM (3F) RMLK 311
TRANNUM (3F) RMUW 331
TRANNUM (C) RMUW 336
transaction
    security domain transaction data, XSXD 455
    security domain transaction token, XSXT 456
    transaction manager catalog records, XMCAT 438
    transaction manager domain anchor block, XMANC 435
    transaction manager resource lock element, XMRLC 440
    transaction manager tran. browse element, XMXBC 441
    transaction manager transaction class, XMCLC 439
    transaction manager transaction definition, XMXDC 441
    transaction manager transaction, XMXNC 445
    user domain transaction data, USXD 410
    user domain transaction token, USXT 411
TRANSACTION_MONITORING_AREA (0) MNCBS 262
TRANSACTION_STG_PTR 294
TRANSIENT_FLAGS (0) BAACT 12, 18
TRANSIENT_FLAGS (4) BAACT 6, 7
TRANSIENT_OBJECT_FACTORY (10) BAACT 16
TRANSIENT_PTR (14) BAACT 5
TRANSIENT_PTR (1C) BAACT 10
TRANSIENT_STATE (0) BAACT 6, 12
TRDM_ACQUIRE 2 LDCBS 184
TRDM_ADD_APE_CELL_POOL_FAIL 2 LDCBS 184
TRDM_ADD_CDE_POOL_FAIL 2 LDCBS 185
TRDM_ADD_CONTROL_POOL_FAIL 2 LDCBS 184
TRDM_ADD_CPE_POOL_FAIL 2 LDCBS 184
TRDM_ADD_CSECTL_POOL_FAIL 2 LDCBS 184
TRDM_ADD_GATE 2 LDCBS 184
TRDM_ADD_LDENRS_POOL_FAIL 2 LDCBS 185
TRDM_ADD_LDENRSRO_POOL_FAIL 2 LDCBS 185
TRDM_ADD_LDENUC_POOL_FAIL 2 LDCBS 184
TRDM_ADD_LDENUCRO_POOL_FAIL 2 LDCBS 184
TRDM_ADD_LDEPGM_POOL_FAIL 2 LDCBS 184
TRDM_ADD_LDEPGMRO_POOL_FAIL 2 LDCBS 185
TRDM_ADD_LDERES_POOL_FAIL 2 LDCBS 184
TRDM_ADD_LDERESRO_POOL_FAIL 2 LDCBS 184
TRDM_ADD_LDNRS_POOL_FAIL 2 LDCBS 185
TRDM_ADD_LDNRSRO_POOL_FAIL 2 LDCBS 185
TRDM_ADD_LDNUC_POOL_FAIL 2 LDCBS 184
TRDM_ADD_LDNUCRO_POOL_FAIL 2 LDCBS 184
TRDM_ADD_LDPGM_POOL_FAIL 2 LDCBS 184
TRDM_ADD_LDPGMRO_POOL_FAIL 2 LDCBS 185
TRDM_ADD_LDRES_POOL_FAIL 2 LDCBS 184
TRDM_ADD_LDRESRO_POOL_FAIL 2 LDCBS 184
TRDM_ADD_LOCK 2 LDCBS 185
TRDM_ADD_LOCK_1 2 LDCBS 185
TRDM_BAD_CC_LOB 2 LDCBS 184
TRDM_CC_WRITE 2 LDCBS 184
TRDM_DEFINE 2 LDCBS 184
TRDM_ENTRY_TRACE 2 LDCBS 183
TRDM_EXIT_TRACE 2 LDCBS 183
TRDM_GET_PARMS 2 LDCBS 184
TRDM_GETMAIN 2 LDCBS 184
TRDM_INQUIRE_START 2 LDCBS 185
TRDM_INVALID_FORMAT 2 LDCBS 183
TRDM_INVALID_FUNCTION 2 LDCBS 183
TRDM_INVALID_PARAMETERS 2 LDCBS 184
TRDM_RECOVERY_ENTERED 2 LDCBS 183
TRDM_RELEASE 2 LDCBS 184
TRDM_SET_ANCHOR 2 LDCBS 185
TRDM_SET_ANCHOR_1 2 LDCBS 185
TRDM_SVC_CALL 2 LDCBS 183
TRDM_SVC_EXCEPTION 2 LDCBS 183
TRDM_SVC_RETURN 2 LDCBS 183
TRDM_UNLOCK 2 LDCBS 185
TRDM_UNLOCK_1 2 LDCBS 185
TRDMI_ADD_GATE 2 LDCBS 187
TRDMI_ADD_GATE_1 2 LDCBS 187
TRDMI_ADD_SUSPEND 2 LDCBS 187
TRDMI_APE_GETMAIN 2 LDCBS 186
TRDMI_BAD_PDB 2 LDCBS 186
TRDMI_BLDL_GETMAIN 2 LDCBS 186
TRDMI_CPE_GETMAIN 2 LDCBS 186
TRDMI_CSECTL_GETMAIN 2 LDCBS 187
TRDMI_CSVQUERY_EXCEPTION 2 LDCBS 186
TRDMI_DELETE_SUSPEND_FAIL 2 LDCBS 187
TRDMI_DFHLDNT 2 LDCBS 186
TRDMI_DFHLDST 2 LDCBS 186
TRDMI_DFHSIP_NOT_FOUND 2 LDCBS 186
TRDMI_END_BROWSE 2 LDCBS 186
TRDMI_GET_PARMS 2 LDCBS 186
TRDMI_GLOBAL_CATALOG 2 LDCBS 186
TRDMI_INQUIRE_START 2 LDCBS 187
TRDMI_LDWE_GETMAIN 2 LDCBS 187
TRDMI_LIBRARY_LOCK 2 LDCBS 186
TRDMI_LIBRARY_UNLOCK 2 LDCBS 186
TRDMI_LIBRARY_UNLOCK_2 2 LDCBS 186
TRDMI_LOCAL_CATALOG 2 LDCBS 186
TRDMI_MODE_CHANGE 2 LDCBS 187
TRDMI_POST_CSVQUERY 2 LDCBS 185
TRDMI_PRE_CSVQUERY 2 LDCBS 185
TRDMI_RECOVERY_ENTERED 2 LDCBS 185
TRDMI_START_BROWSE 2 LDCBS 186
TRDMI_STATE_LOCK 2 LDCBS 187
TRDMI_STATE_LOCK_1 2 LDCBS 187
TRDMI_STATE_LOCK_2 2 LDCBS 187
TRDMI_STATE_LOCK_3 2 LDCBS 187
TRDMI_STATE_LOCK_4 2 LDCBS 187
TRDMI_STATE_LOCK_5 2 LDCBS 187
TRDMI_STATE_LOCK_6 2 LDCBS 187
TRDMI_STATE_UNLOCK 2 LDCBS 187
TRDMI_STATE_UNLOCK_1 2 LDCBS 187
TRDMI_STATE_UNLOCK_2 2 LDCBS 187
TRDMI_STATE_UNLOCK_3 2 LDCBS 187

TRDMI_STATE_UNLOCK_4 2 LDCBS 187
TRDMI_STATE_UNLOCK_5 2 LDCBS 187
TRDMI_STATE_UNLOCK_6 2 LDCBS 187
TRDMI_STATE_UNLOCK_7 2 LDCBS 187
TRDMI_STATE_UNLOCK_8 2 LDCBS 187
TRDMI_STATE_UNLOCK_9 2 LDCBS 187
TRDMI_SUSPEND_FAIL 2 LDCBS 187
TRDMI_SVC_CALL 2 LDCBS 185
TRDMI_SVC_EXCEPTION 2 LDCBS 185
TRDMI_SVC_RETURN 2 LDCBS 185
TRDMI_TYPE_PURGE 2 LDCBS 187
TRDMI_WAIT_PHASE 2 LDCBS 186
TRE_ALARM_CALL (BIT) TIA 379
TRE_ALARM_TIME (34) TIA 379
TRE_ARROW (2) TIA 379
TRE_ATTACH_MODE (4B) TIA 379
TRE_ATTACH_PRIORITY (4A) TIA 379
TRE_ATTACH_TIMEOUT (4C) TIA 379
TRE_ATTACHED_TASK (BIT) TIA 379
TRE_BLOCK_NAME (8) TIA 379
TRE_CANCELLED (BIT) TIA 379
TRE_CO (BIT) TIA 379
TRE_DFH (3) TIA 379
TRE_DOMAIN_ID (18) TIA 379
TRE_DOMAIN_TOKEN (1C) TIA 379
TRE_DOMID (6) TIA 379
TRE_EXPIRED (BIT) TIA 379
TRE_EXPIRY_TIME (24) TIA 379
TRE_EXPIRY_TIME_HIGH (24) TIA 379
TRE_EXPIRY_TIME_LOW (28) TIA 379
TRE_FLAGS (49) TIA 379
TRE_FO (BIT) TIA 379
TRE_INTERVAL (2C) TIA 379
TRE_INTERVAL_MSECS (30) TIA 379
TRE_INTERVAL_NOTIFY (BIT) TIA 379
TRE_INTERVAL_SECS (2C) TIA 379
TRE_LENGTH (0) TIA 379
TRE_NEXT (10) TIA 379
TRE_NOTIFY_TYPE (48) TIA 379
TRE_NUMBER (50) TIA 379
TRE_ORIGIN_DATE (40) TIA 379
TRE_ORIGIN_INTERVAL_EXPIRED (BIT) TIA 379
TRE_ORIGIN_TIME (3A) TIA 379
TRE_PERIODIC (BIT) TIA 379
TRE_PREFIX (0) TIA 379
TRE_PREV (14) TIA 379
TRE_QR (BIT) TIA 379
TRE_RESET_TIME_PROCESSED (BIT) TIA 379
TRE_RO (BIT) TIA 379
TRE_TIMER_TASK (BIT) TIA 379
TRE_WITH_ATTMODE (BIT) TIA 379
TRE_WITH_ORIGIN (BIT) TIA 379
TRE_WITH_TIMEOUT (BIT) TIA 379
TRID_CC_ADD_LEN 2 CCGD 31
TRID_CC_CHANGE_MODE 2 CCGD 31
TRID_CC_DATA_TOO_LONG 2 CCGD 31
TRID_CC_ENTRY 2 CCGD 31
TRID_CC_EXIT 2 CCGD 31
TRID_CC_EXTENT 2 CCGD 31
TRID_CC_FUNCTION 2 CCGD 31
TRID_CC_NOT_FOR_LCD 2 CCGD 31
TRID_CC_PUT_R_LEN 2 CCGD 31
TRID_CC_RECOVERY 2 CCGD 31
TRID_CC_RESTORE_MODE 2 CCGD 31
TRID_CC_SERIAL_ENTRY 2 CCGD 31
TRID_CC_SERIAL_EXIT 2 CCGD 31
TRID_CC_ST_WAIT_LOCK 2 CCGD 31
TRID_CC_ST_WAIT_UNLOCK 2 CCGD 31
TRID_CC_TOKEN 2 CCGD 31
TRID_CC_TOKEN2 2 CCGD 31
TRID_CC_TOKEN3 2 CCGD 31
TRID_CC_TOKEN4 2 CCGD 31
TRID_CC_TOKEN5 2 CCGD 31
TRID_CC_TOKEN6 2 CCGD 31
TRID_CC_TOKEN7 2 CCGD 31
TRID_CC_TOKEN8 2 CCGD 31
TRID_CC_TOKEN9 2 CCGD 31
TRID_CC_USE_TOKEN 2 CCGD 31
TRID_CC_USE_WRITE_N 2 CCGD 31
TRID_CC_VSAM 2 CCGD 31
TRID_CC_VSAM_END 2 CCGD 31
TRID_CC_VSAM_WAIT 2 CCGD 31
TRID_CC_WAIT_OLDC 2 CCGD 31
TRID_CC_WR_NX_LEN 2 CCGD 31
TRID_CC_WRITE_LEN 2 CCGD 31
TRID_CC_XC_WAIT_LOCK 2 CCGD 31
TRID_CC_XC_WAIT_UNLOCK 2 CCGD 31
TRID_DM_ADD_LOCK 2 CCGD 31
TRID_DM_ENTRY 2 CCGD 31
TRID_DM_EXIT 2 CCGD 31
TRID_DM_RECOVERY 2 CCGD 31
TRID_DM_SET_PHASE 2 CCGD 31
TRID_DM_UNLOCK 2 CCGD 31
TRID_DM_VSAM_ERROR 2 CCGD 31
TRIM_CHAIN_HEADER (0) L2LF 234
TRLD_ADD_SUSPEND 2 LDCBS 178
TRLD_BAD_PDB 2 LDCBS 178
TRLD_CPE_GETMAIN 2 LDCBS 178
TRLD_DELETE_SUSPEND 2 LDCBS 178
TRLD_ENTRY_TRACE 2 LDCBS 177
TRLD_EXIT_TRACE 2 LDCBS 177
TRLD_INQUIRE_START 2 LDCBS 179
TRLD_INVALID_ENTRY_POINT 2 LDCBS 178
TRLD_INVALID_FORMAT 2 LDCBS 178
TRLD_INVALID_FUNCTION 2 LDCBS 178
TRLD_INVALID_PARAMETERS 2 LDCBS 178
TRLD_INVALID_PGM_TOKEN 2 LDCBS 178
TRLD_INVALID_PGM_TOKEN_1 2 LDCBS 178
TRLD_INVALID_PGM_TOKEN_2 2 LDCBS 178
TRLD_LDWE_GETMAIN 2 LDCBS 178
TRLD_LOCK 2 LDCBS 178
TRLD_LOCK_1 2 LDCBS 178
TRLD_RECOVERY_ENTERED 2 LDCBS 178
TRLD_SUSPEND 2 LDCBS 178
TRLD_UNLOCK 2 LDCBS 179
TRLD_UNLOCK_1 2 LDCBS 179
TRLD1_APE_GETMAIN 2 LDCBS 180
TRLD1_BAD_STRUCTURE 2 LDCBS 180
TRLD1_CDE_GETMAIN_FAIL 2 LDCBS 180
TRLD1_CSECTL_GETMAIN 2 LDCBS 180
TRLD1_CSVQUERY_EXCEPTION 2 LDCBS 180
TRLD1_DSA_COMPRESSION 2 LDCBS 179
TRLD1_INVALID_FUNCTION 2 LDCBS 179
TRLD1_LIBRARY_IO_ERROR 2 LDCBS 181
TRLD1_LIBRARY_IO_ERROR_1 2 LDCBS 181
TRLD1_LIBRARY_LOCK 2 LDCBS 181
TRLD1_LIBRARY_LOCK_1 2 LDCBS 181
TRLD1_LIBRARY_LOCK_2 2 LDCBS 181
TRLD1_LIBRARY_LOCK_3 2 LDCBS 181
TRLD1_LIBRARY_UNLOCK 2 LDCBS 181
TRLD1_LIBRARY_UNLOCK_1 2 LDCBS 181
TRLD1_LIBRARY_UNLOCK_2 2 LDCBS 181
TRLD1_LIBRARY_UNLOCK_3 2 LDCBS 181
TRLD1_LIBRARY_UNLOCK_4 2 LDCBS 181
TRLD1_MODE_CHANGE 2 LDCBS 181
TRLD1_MODE_CHANGE_1 2 LDCBS 181
TRLD1_MODE_CHANGE_2 2 LDCBS 181
TRLD1_NO_OS_STORAGE 2 LDCBS 181
TRLD1_NO_OS_STORAGE_1 2 LDCBS 181
TRLD1_PGM_GETMAIN 2 LDCBS 180
TRLD1_POST_CSVQUERY 2 LDCBS 179
TRLD1_PRE_CSVQUERY 2 LDCBS 179
TRLD1_RECOVERY_ENTERED 2 LDCBS 179
TRLD1_STATE_LOCK 2 LDCBS 180
TRLD1_STATE_LOCK_1 2 LDCBS 180
TRLD1_STATE_LOCK_2 2 LDCBS 180
TRLD1_STATE_LOCK_3 2 LDCBS 180
TRLD1_STATE_LOCK_4 2 LDCBS 180
TRLD1_STATE_LOCK_5 2 LDCBS 180
TRLD1_STATE_LOCK_6 2 LDCBS 180
TRLD1_STATE_UNLOCK 2 LDCBS 180
TRLD1_STATE_UNLOCK_1 2 LDCBS 180
TRLD1_STATE_UNLOCK_2 2 LDCBS 180
TRLD1_STATE_UNLOCK_3 2 LDCBS 180
TRLD1_STATE_UNLOCK_4 2 LDCBS 181
TRLD1_SVC_CALL 2 LDCBS 179
TRLD1_SVC_EXCEPTION 2 LDCBS 180
TRLD1_SVC_REQUEST_FAILURE 2 LDCBS 181
TRLD1_SVC_REQUEST_FAILURE_1 2 LDCBS 181
TRLD1_SVC_RETURN 2 LDCBS 179
TRLD2_CC_DELETE 2 LDCBS 181
TRLD2_CC_WRITE 2 LDCBS 181
TRLD2_CC_WRITE_2 2 LDCBS 181
TRLD2_CPE_GETMAIN 2 LDCBS 181
TRLD2_RECOVERY_ENTERED 2 LDCBS 181
TRLD2_SVC_CALL 2 LDCBS 179

TRLD2_SVC_EXCEPTION 2 LDCBS 180
TRLD2_SVC_RETURN 2 LDCBS 179
TRLD3_CC_WRITE 2 LDCBS 182
TRLD3_CC_WRITE_PDB1 2 LDCBS 182
TRLD3_CC_WRITE_PDB2 2 LDCBS 182
TRLD3_CC_WRITE_PDB3 2 LDCBS 182
TRLD3_CC_WRITE_PDB4 2 LDCBS 182
TRLD3_LDBE_GETMAIN 2 LDCBS 182
TRLD3_LIBRARY_LOCK 2 LDCBS 181
TRLD3_LIBRARY_LOCK_1 2 LDCBS 181
TRLD3_LIBRARY_UNLOCK 2 LDCBS 181
TRLD3_LIBRARY_UNLOCK_1 2 LDCBS 181
TRLD3_LONG_NAME 2 LDCBS 180
TRLD3_MODE_CHANGE 2 LDCBS 180
TRLD3_PRVMOD_GETMAIN 2 LDCBS 182
TRLD3_RECOVERY_ENTERED 2 LDCBS 182
TRLD3_SVC_CALL 2 LDCBS 179
TRLD3_SVC_EXCEPTION 2 LDCBS 180
TRLD3_SVC_RETURN 2 LDCBS 179
TRN_DB2ENTRY_ADDR (24) D2TRN 98
TRN_DB2ENTRY_COUNT (28) D2TRN 98
TRN_DB2ENTRY_ETOKEN (24) D2TRN 98
TRN_DB2ENTRY_NAME (1C) D2TRN 98
TRN_EYE (2) D2TRN 98
TRN_LENGTH (0) D2TRN 98
TRN_NAME (10) D2TRN 98
TRN_PREFIX (0) D2TRN 98
TRN_TRANSID (18) D2TRN 98
TRNT_ENTRY_TRACE 2 LDCBS 182
TRNT_EXIT_TRACE 2 LDCBS 182
TRNT_INVALID_FORMAT 2 LDCBS 182
TRNT_INVALID_FUNCTION 2 LDCBS 182
TRNT_INVALID_PARAMETERS 2 LDCBS 182
TRNT_LOCK_FAILURE 2 LDCBS 182
TRNT_RECOVERY_ENTERED 2 LDCBS 182
TRNT_UNLOCK_FAILURE 2 LDCBS 182
TRRESPONSE (0) CAUTR 28
TRST_ENTRY_TRACE 2 LDCBS 182
TRST_EXIT_TRACE 2 LDCBS 182
TRST_INVALID_FORMAT 2 LDCBS 183
TRST_INVALID_FUNCTION 2 LDCBS 183
TRST_INVALID_PARAMETERS 2 LDCBS 183
TRST_LOCK_FAILURE 2 LDCBS 183
TRST_RECOVERY_ENTERED 2 LDCBS 183
TRST_UNLOCK_FAILURE 2 LDCBS 183
TRUE 391
TRUE 0 CCGD 31
TRUE 0 DDCBC 38
TRUE 0 STUCB 378
TRUNCATE 430
TRUNCATE_NO 1 WRB 430
TRUNCATE_YES 1 WRB 430
TSA 380
TSA (0) TSA 380
TSA_AGING_TIME (60) TSA 380
TSA_ARROW (2) TSA 380
TSA_BLOCK_NAME (8) TSA 380
TSA_BUFFERS (48) TSA 380
TSA_DFH (3) TSA 380
TSA_DOMID (6) TSA 380
TSA_FLAGS (3A) TSA 380
TSA_LAST_COLD_START_TIME (40) TSA 380
TSA_LENGTH (0) TSA 380
TSA_MAIN_ONLY (BIT) TSA 380
TSA_PREFIX (0) TSA 380
TSA_RDO_ENABLED (BIT) TSA 380
TSA_SHARED_ANCHORP (58) TSA 380
TSA_START (39) TSA 380
TSA_START_AUTO 4 TSA 381
TSA_START_COLD 4 TSA 381
TSA_START_EMERGENCY 4 TSA 381
TSA_START_WARM 4 TSA 381
TSA_STATS_RESET_TIME (50) TSA 380
TSA_STRINGS (4C) TSA 380
TSA_SYSID_TABLE_TOKEN (5C) TSA 380
TSA_TS_STATE (38) TSA 380
TSA_TSAUX_CLASSP (34) TSA 380
TSA_TSGENRAL_SPTOKEN (10) TSA 380
TSA_TSLOCK (30) TSA 380
TSA_TSMAIN_CLASSP (20) TSA 380
TSA_TSMODEL_CLASSP (68) TSA 380
TSA_TSNAME_CLASSP (18) TSA 380
TSA_TSOLOCK_CLASSP (28) TSA 380
TSA_TSQUEUE_CLASSP (1C) TSA 380
TSA_TSRLOCK_CLASSP (2C) TSA 380
TSA_TSTP (3C) TSA 380
TSA_TSWAITQ_CLASSP (24) TSA 380
TSA_XRSINDI_ACTIVE (BIT) TSA 380
TSA_XTSPTIN_ACTIVE (BIT) TSA 380
TSA_XTSPTOUT_ACTIVE (BIT) TSA 380
TSA_XTSQRIN_ACTIVE (BIT) TSA 380
TSA_XTSQROUT_ACTIVE (BIT) TSA 380
TSAUX 384
TSAUX (0) TSAUX 384
tsf
    tsf - eye catcher map, FEP09 131
TSH_BROWSE_END 4 TSRL 400
TSH_DISASTER 4 TSRL 400
TSH_NOT_FOUND 4 TSRL 400
TSH_OK 4 TSRL 400
TSH_PURGED 4 TSRL 400
TSH_RESPONSE 400
TSI 397
TSI_ITEMT (4) TSQU 397
TSI_NEXT (0) TSQU 397
TSIOA (0) TSAUX 388
TSIOA (0) TSMN 392
TSIOA_EYECATCHER (0) TSAUX 388
TSIOA_EYECATCHER (0) TSMN 392
TSIOA_EYECATCHER_STRING 8 TSAUX 389
TSIOA_EYECATCHER_STRING 8 TSMN 393
TSLOCK_NAME 8 TSA 381
TSM 392
TSM_CLASS_ANCHOR 392
TSM_CURV (8) TSMN 392
TSM_DATA (8) TSMN 392
TSM_DISASTER 4 TSMN 393
TSM_EYECATCHER (0) TSMN 392
TSM_EYECATCHER_VALUE 4 TSMN 393
TSM_FIXED_LENGTH_TAB (0) TSMN 392
TSM_FLAGS (4) TSMN 392
TSM_FMH 392
TSM_INVALID_EYECATCHER 4 TSMN 393
TSM_LENGTH (6) TSMN 392
TSM_MAXV (C) TSMN 392
TSM_NMG (4) TSMN 392
TSM_NMP (0) TSMN 392
TSM_OK 4 TSMN 393
TSM_PREFIX (0) TSMN 392
TSM_PURGED 4 TSMN 393
TSM_RESPONSE (0) TSMN 392
TSM_SPPREFIX 4 TSMN 393
TSM_SPTOKEN (10) TSMN 392
TSM_SUFFIX_TAB (10) TSMN 392
TSMAIN (0) TSMN 392
TSMD_MODEL_TYPE 391
TSMD_RDO_DISABLED 391
TSMD_RDO_ENABLED 391
TSMD_RDO_NAME 391
TSMD_RDO_TYPE 391
TSMN 390, 392
TSMODEL (0) TSMN 390
TSMODELNAME 391
TSN_BRB_FIRST (30) TSNM 393
TSN_BRB_LAST (34) TSNM 393
TSN_BRB_SPTOKEN (18) TSNM 393
TSN_BRBHEAD (30) TSNM 393
TSN_CHANGE_COUNT (2C) TSNM 393
TSN_CLASS_ANCHOR 393
TSN_DISASTER 4 TSNM 394
TSN_DTN_SPTOKEN 393
TSN_DUPLICATE 4 TSNM 394
TSN_END_BROWSE 4 TSNM 394
TSN_INVALID_NAME 4 TSNM 394
TSN_INVALID_PREFIX 4 TSNM 394
TSN_NOT_FOUND 4 TSNM 394
TSN_NQCR (28) TSNM 393
TSN_OK 4 TSNM 394
TSN_PURGED 4 TSNM 394
TSN_QNUM 393
TSN_QNUMH (24) TSNM 393
TSN_RESPONSE 394
TSN_ROOTP (0) TSNM 393
TSN_TSQ_SPTOKEN (10) TSNM 393
TSNAME (0) TSNM 393
TSNM 393

TSO_CLASS_ANCHOR 395
TSO_DISASTER 4 TSOL 396
TSO_KEYPT_BUFFER (34) TSOL 395
TSO_KEYPT_BUFFER_HEADER 395
TSO_KEYPT_BUFFER_LENGTH 4 TSOL 396
TSO_LOCKED 4 TSOL 396
TSO_NQTOKEN (10) TSOL 395
TSO_OK 4 TSOL 396
TSO_PURGED 4 TSOL 396
TSO_QAB_FIRST (18) TSOL 395
TSO_QAB_LAST (1C) TSOL 395
TSO_QAB_SPTOKEN (0) TSOL 395
TSO_QABHEAD 395
TSO_QOB_SPTOKEN (8) TSOL 395
TSO_QOBP (0) TSOL 394
TSO_QOBP (38) TSQU 396
TSO_RESPONSE 395
TSO_RESTART 4 TSOL 396
 TSOL 394
TSOLOCK (0) TSOL 394
TSPREFIX (0) TSMN 391
TSQ (0) TSQU 396
TSQ_BMS (BIT) TSQU 397, 398
TSQ_CHECK_FAILED 4 TSQU 398
TSQ_CLASS_ANCHOR (0) TSQU 397
TSQ_COMMITTED_ITEMS (3C) TSQU 396
TSQ_CREATION_TIME (48) TSQU 396
TSQ_DELETE_SEEN (BIT) TSQU 397, 398
TSQ_DELETED (BIT) TSQU 397, 398
TSQ_DISASTER 4 TSQU 398
TSQ_DISCARD 397, 398
TSQ_DUPLICATE_NAME 4 TSQU 398
TSQ_FIRST_OPERATION (62) TSQU 397
TSQ_FIRST_TSIP (14) TSQU 396
TSQ_FLAG_BYTES 397
TSQ_FLAGS (60) TSQU 397
TSQ_FULL 4 TSQU 398
TSQ_IC (BIT) TSQU 397, 398
TSQ_IC_DATA_N (1C) TSQU 397
TSQ_IC_DATA_P (5C) TSQU 397
TSQ_IC_SPTOKEN (10) TSQU 397
TSQ_INVALID_LENGTH 4 TSQU 398
TSQ_INVALID_TYPE 4 TSQU 398
TSQ_ITEM_NOT_FOUND 4 TSQU 398
TSQ_LAST_REFERENCED_TIME (50) TSQU 397
TSQ_LAST_TSIP (18) TSQU 396
TSQ_LOCKED 4 TSQU 398
TSQ_MAIN 397, 398
TSQ_NAME (0) TSQU 396
TSQ_NEW (BIT) TSQU 397, 398
TSQ_NOSPACE 4 TSQU 398
TSQ_OK 4 TSQU 398
TSQ_OLD_CREATION_TIME (68) TSQU 397
TSQ_OLD_IC_DATA_P 397
TSQ_OPERATION_GET_RELEASE 4 TSQU 398
TSQ_OPERATION_NULL 4 TSQU 398
TSQ_OPERATION_PUT 4 TSQU 398
TSQ_OPERATION_RELEASE 4 TSQU 398
TSQ_OWNED (BIT) TSQU 397, 398
TSQ_OWNERSHIP_LOCK (38) TSQU 396
TSQ_PREFIX (0) TSQU 396
TSQ_PURGED 4 TSQU 398
TSQ_PUT (BIT) TSQU 397, 398
TSQ_QINH (20) TSQU 397
TSQ_QUB_FIRST (40) TSQU 396
TSQ_QUB_LAST (44) TSQU 396
TSQ_QUB_SPTOKEN (8) TSQU 397
TSQ_QUBHEAD (40) TSQU 396
TSQ_QUEUE_DELETED 4 TSQU 398
TSQ_READ_CURSOR (20) TSQU 396
TSQ_READ_TSIP (24) TSQU 396
TSQ_RECOVERABLE (BIT) TSQU 397, 398
TSQ_REQUEST_LOCK (28) TSQU 396
TSQ_RESPONSE (0) TSQU 398
TSQ_REST (14) TSQU 396
TSQ_RESTART 4 TSQU 398
TSQ_SHUNTED (BIT) TSQU 397, 398
TSQ_TOTAL_ITEMS (1C) TSQU 396
TSQ_TRANSID (58) TSQU 397
TSQ_TSI_SPTOKEN (0) TSQU 397
TSQ_TSIFREEHEAD (18) TSQU 397
TSQ_UP (10) TSQU 396
 TSQU 396
TSQUEUE (0) TSQU 396
TSR_CLASS_ANCHOR (0) TSRL 401
TSR_DELETED 4 TSRL 401
TSR_DISASTER 4 TSRL 401
TSR_OK 4 TSRL 401
TSR_OWNER (30) TSQU 396
TSR_OWNER (8) TSRL 401
TSR_PURGED 4 TSRL 401
TSR_RESPONSE (0) TSRL 401
TSR_RESTART 4 TSRL 401
TSR_WAITQ (0) TSRL 401
TSR_WAITQ (28) TSQU 396
 TSRL 399, 401
TSRLOCK (0) TSRL 401
TSS (0) TSAUX 388
TSS_CI_NUMBER (4) TSAUX 388
TSS_NEXT (0) TSAUX 388
TSS_SECTION_LENGTH (6) TSAUX 388
TSSHARED (0) TSRL 399
TSSYSID (0) TSMN 391
TSW (0) TSWQ 402
TSW_AUX_SPACE 1 TSWQ 403
TSW_BUFFER 1 TSWQ 403
TSW_CLASS_ANCHOR 402
TSW_DISASTER 4 TSWQ 403
TSW_EXTEND 1 TSWQ 403
TSW_FIRST (0) TSRL 401
TSW_FIRST (0) TSWQ 402
TSW_FIRST (10) TSRL 400
TSW_FIRST (18) TSOL 395
TSW_FIRST (28) TSAUX 384
TSW_FIRST (28) TSQU 396
TSW_FIRST (30) TSAUX 384
TSW_FIRST (38) TSAUX 384
TSW_FIRST (40) TSAUX 384
TSW_FIRST (48) TSAUX 384
TSW_FLAGS (1C) TSWQ 402
TSW_HEAD (0) TSRL 401
TSW_HEAD (0) TSWQ 402
TSW_HEAD (10) TSRL 400
TSW_HEAD (18) TSOL 395
TSW_HEAD (28) TSAUX 384
TSW_HEAD (28) TSQU 396
TSW_HEAD (30) TSAUX 384
TSW_HEAD (38) TSAUX 384
TSW_HEAD (40) TSAUX 384
TSW_HEAD (48) TSAUX 384
TSW_LAST (14) TSRL 400
TSW_LAST (1C) TSOL 395
TSW_LAST (2C) TSAUX 384
TSW_LAST (2C) TSQU 396
TSW_LAST (34) TSAUX 384
TSW_LAST (3C) TSAUX 384
TSW_LAST (4) TSRL 401
TSW_LAST (4) TSWQ 402
TSW_LAST (44) TSAUX 384
TSW_LAST (4C) TSAUX 384
TSW_NEXT (0) TSWQ 402
TSW_OK 4 TSWQ 403
TSW_POOL 1 TSWQ 403
TSW_PREFIX (0) TSWQ 402
TSW_PREV (4) TSWQ 402
TSW_PURGED 4 TSWQ 403
TSW_QUEUE 1 TSWQ 403
TSW_RESOURCE_TYPE 402
TSW_RESPONSE 402
TSW_RESTART 4 TSWQ 403
TSW_RESTART_REQUIRED (BIT) TSWQ 402
TSW_RESTYPE (0) TSWQ 402
TSW_RESUME_PRIORITY (1E) TSWQ 402
TSW_STRING 1 TSWQ 403
TSW_SUSPEND_START_TIME (10) TSWQ 402
TSW_SUSPEND_TOKEN (8) TSWQ 402
TSW_TRANSACTION_NUMBER (18) TSWQ 402
TSW_TSW_SPTOKEN (0) TSWQ 402
TSW_WAITER (C) TSWQ 402
TSW_WRITE_BUFFER 1 TSWQ 403
TSWAITQ (0) TSWQ 402
 TSWQ 402
TSX 388
TSX_CHECK_FAILED 4 TSAUX 389
TSX_CLOSE_FAILED 4 TSAUX 389
TSX_DATASET_EMPTY 4 TSAUX 389

TSX_DISASTER 4 TSAUX 389
TSX_NO_CONTROL_RECORD 4 TSAUX 389
TSX_NOSPACE 4 TSAUX 389
TSX_OK 4 TSAUX 389
TSX_OPEN_FAILED 4 TSAUX 389
TSX_PURGED 4 TSAUX 389
TSX_RESPONSE (0) TSAUX 389
TSX_SHOWCB_FAILED 4 TSAUX 389
TSX_TIME_STAMP (0) TSAUX 388
TSX_TOTAL_LENGTH (8) TSAUX 388
TSX_TSSP (C) TSAUX 388
TUNING_INTERVAL 4 SMDCC 363
TURN_OFF_LAST_3_BITS 4 PAA 284
TXD_INSTANCE (0) XMXDC 441
TXD_STATIC (0) XMXDC 443
TXDINST_ADD_CREATED (BIT) XMXDC 441
TXDINST_AP_TOKEN (34) XMXDC 441
TXDINST_ARROW (2) XMXDC 441
TXDINST_BACK_CHAIN (18) XMXDC 441
TXDINST_BLOCK_NAME (8) XMXDC 441
TXDINST_BREXIT 443
TXDINST_COMMAND_SECURITY (8F) XMXDC 442
TXDINST_CONFDATA (8D) XMXDC 442
TXDINST_DFH (3) XMXDC 441
TXDINST_DOMID (6) XMXDC 441
TXDINST_DTIMEOUT (90) XMXDC 442
TXDINST_DTRTRAN (BIT) XMXDC 441
TXDINST_DYNAMIC (A8) XMXDC 442
TXDINST_EXTERNAL_FLAGS (BB) XMXDC 442
TXDINST_EXTERNALS (60) XMXDC 442
TXDINST_INDOUBT_ACTION (83) XMXDC 442
TXDINST_INDOUBT_WAIT (82) XMXDC 442
TXDINST_INDOUBT_WAIT_TIME (84) XMXDC 442
TXDINST_INITIAL_PROGRAM (60) XMXDC 442
TXDINST_INSTANCE_ADDR (1C) XMXDC 441
TXDINST_INSTANCE_NUMBER (20) XMXDC 441
TXDINST_ISOLATED_SUBSPACE (BA) XMXDC 442
TXDINST_LENGTH (0) XMXDC 441
TXDINST_LOCAL_QUEUING (A9) XMXDC 442
TXDINST_MISCELLANEOUS_FLAGS (28) XMXDC 441
TXDINST_PARTITIONSET (77) XMXDC 442
TXDINST_PARTITIONSET_NAME (78) XMXDC 442
TXDINST_PG_TOKEN 442
TXDINST_PREFIX (0) XMXDC 441
TXDINST_PROFILE_NAME (68) XMXDC 442
TXDINST_REMOTE 441
TXDINST_REMOTE_NAME (94) XMXDC 442
TXDINST_REMOTE_SYSTEM (9C) XMXDC 442
TXDINST_REMOTE_SYSTEM_SPECIFIED (BIT) XMXDC 443
TXDINST_RESOURCE_SECURITY (8E) XMXDC 442
TXDINST_RESTART (B4) XMXDC 442
TXDINST_ROUTABLE_STATUS (C4) XMXDC 443
TXDINST_RUNAWAY_LIMIT (88) XMXDC 442
TXDINST_SET_CREATED (BIT) XMXDC 441
TXDINST_SHUTDOWN_OVERRIDE (BIT) XMXDC 441
TXDINST_SHUTDOWN_STATUS (B9) XMXDC 442
TXDINST_STATIC_BLOCK_ADDR (14) XMXDC 441
TXDINST_STATUS (80) XMXDC 442
TXDINST_STORAGE_CLEAR (8C) XMXDC 442
TXDINST_STORAGE_FREEZE (AA) XMXDC 442
TXDINST_SYSTEM_ATTACH 441
TXDINST_SYSTEM_PURGEABLE (B5) XMXDC 442
TXDINST_SYSTEM_RUNAWAY (81) XMXDC 442
TXDINST_TASKDATAKEY (74) XMXDC 442
TXDINST_TASKDATALOC (75) XMXDC 442
TXDINST_TCLASS (AB) XMXDC 442
TXDINST_TCLASS_NAME (AC) XMXDC 442
TXDINST_TCLASS_TOKEN 441
TXDINST_TERMERR_PURGEABLE (B6) XMXDC 442
TXDINST_TRAN_PRIORITY (76) XMXDC 442
TXDINST_TRANDEF_RELATED_TOKENS (34) XMXDC 441
TXDINST_TRANDEF_TOKEN (1C) XMXDC 441
TXDINST_TRANSACTION_DUMP (B7) XMXDC 442
TXDINST_TRANSACTION_ID (10) XMXDC 441
TXDINST_TRANSACTION_TRACE (B8) XMXDC 442
TXDINST_TRPROF (A0) XMXDC 442
TXDINST_TWASIZE (70) XMXDC 442
TXDINST_USE_COUNT (24) XMXDC 441
TXDSTAT_ACTION_MISMATCHES (78) XMXDC 444
TXDSTAT_ACTIVE (BIT) XMXDC 443
TXDSTAT_ALIAS (6C) XMCAT 438
TXDSTAT_ALIAS (88) XMXDC 444
TXDSTAT_ALIAS_EXISTENCE_BITS (68) XMCAT 438
TXDSTAT_ALIAS_EXISTENCE_BITS (84) XMXDC 444
TXDSTAT_ALIAS_X (BIT) XMCAT 438
TXDSTAT_ALIAS_X (BIT) XMXDC 444
TXDSTAT_ALIASES 444
TXDSTAT_ARROW (2) XMXDC 443
TXDSTAT_ATTACH_COUNT (48) XMXDC 443
TXDSTAT_BLOCK_NAME (8) XMXDC 443
TXDSTAT_CREATION_TIME (40) XMXDC 443
TXDSTAT_DFH (3) XMXDC 443
TXDSTAT_DOMID (6) XMXDC 443
TXDSTAT_DYN_LOCAL_COUNT (54) XMXDC 443
TXDSTAT_DYN_REMOTE_COUNT (58) XMXDC 443
TXDSTAT_FORCED_ACTN_NOWAIT (60) XMXDC 443
TXDSTAT_FORCED_ACTN_OPERATOR (64) XMXDC 443
TXDSTAT_FORCED_ACTN_OTHER (70) XMXDC 444
TXDSTAT_FORCED_ACTN_TIMEOUT (68) XMXDC 443
TXDSTAT_FORCED_ACTN_TRANDEF (6C) XMXDC 444
TXDSTAT_INDOUBT_WAIT_COUNT (74) XMXDC 444
TXDSTAT_INT_ATTACHES (A0) XMXDC 444
TXDSTAT_INT_TCB_COUNTS (A4) XMXDC 444
TXDSTAT_INTERVAL_COUNTS (A0) XMXDC 444
TXDSTAT_LATEST_INSTANCE (14) XMXDC 443
TXDSTAT_LENGTH (0) XMXDC 443
TXDSTAT_LOCK_TOKEN (2C) XMXDC 443
TXDSTAT_NEXT_DECAY (94) XMXDC 444
TXDSTAT_NEXT_STATIC_BLOCK (18) XMXDC 443
TXDSTAT_PREFIX (0) XMXDC 443
TXDSTAT_REMOTE_DIR_NEXT (28) XMXDC 443
TXDSTAT_REMOTE_DIR_PREV (24) XMXDC 443
TXDSTAT_REMOTE_DIR_X (BIT) XMXDC 443
TXDSTAT_REMOTE_START_COUNT (5C) XMXDC 443
TXDSTAT_RESTART_COUNT (4C) XMXDC 443
TXDSTAT_STATUS_FLAGS (20) XMXDC 443
TXDSTAT_STG_VIOLATIONS (50) XMXDC 443
TXDSTAT_SYSTEM_DEFINITION (BIT) XMXDC 443
TXDSTAT_TASKREQ (70) XMCAT 438
TXDSTAT_TASKREQ (8C) XMXDC 444
TXDSTAT_TASKREQ_X (BIT) XMCAT 438
TXDSTAT_TASKREQ_X (BIT) XMXDC 444
TXDSTAT_TCB_COUNTS (94) XMXDC 444
TXDSTAT_TOT_ATTACHES (98) XMXDC 444
TXDSTAT_TOT_TCB_COUNTS (9C) XMXDC 444
TXDSTAT_TOTAL_COUNTS (98) XMXDC 444
TXDSTAT_TPNAME_ADDR (7C) XMXDC 444
TXDSTAT_TPNAME_X (BIT) XMCAT 438
TXDSTAT_TPNAME_X (BIT) XMXDC 444
TXDSTAT_TRANDEF_STATS 443
TXDSTAT_TRANSACTION_ID (10) XMXDC 443
TXDSTAT_USE_COUNT (1C) XMXDC 443
TXDSTAT_XTRANID (74) XMCAT 438
TXDSTAT_XTRANID (90) XMXDC 444
TXDSTAT_XTRANID_X (BIT) XMCAT 438
TXDSTAT_XTRANID_X (BIT) XMXDC 444
TYPE (10) L2HP 226
TYPE (17) UDB 404
TYPE (1C) SOA 371
TYPE (43) UDB 404
TYPE (68) L2CH 221
TYPE (69) DSTSK 66
TYPE (A0) L2CH 223
TYPE (F) XSSS 455
TYPE_CATALOG (14) CCGD 29
TYPE_OF_STREAM (C7) L2BS 213
TYPE_OF_STREAM (C7) L2SR 244
TYPES_USED (B8) DSTSK 66

U

UB_CHAINING (8) STCB1 375
UB_DATA (AE) STCB1 375
UB_DATA_LEN (4) STCB1 375
UB_LENGTH (0) STCB1 375
UB_NEXT (C) STCB1 375
UB_PREV (8) STCB1 375
UB_SMF_HEADER (10) STCB1 375
UB_SMF_PS (3C) STCB1 375
UDB 403
UDSA 4 SMDCC 363
UDSA_NAME 8 SMDCC 363
UID_LEN (12) BAACT 6, 8
UID_LEN (2) BAACT 7, 13, 17
UID_LEN (22) BAACT 5, 10

UID_LEN (34) BAACT 17
UID_LEN (3C) BAACT 15
UID_LEN (54) BAACT 11
UID_LEN (6) BAACT 14
UID_LEN (7A) BAACT 16
UID_LEN (A) BAACT 14, 15
UID_LEN (AC) BAACT 16
UID_LEN (E) BAACT 12, 18
UID_LU_LEN (13) BAACT 6, 8
UID_LU_LEN (23) BAACT 5, 10
UID_LU_LEN (3) BAACT 7, 13, 17
UID_LU_LEN (35) BAACT 17
UID_LU_LEN (3D) BAACT 15
UID_LU_LEN (55) BAACT 11
UID_LU_LEN (7) BAACT 14
UID_LU_LEN (7B) BAACT 16
UID_LU_LEN (AD) BAACT 16
UID_LU_LEN (B) BAACT 14, 15
UID_LU_LEN (F) BAACT 12, 19
ULT_FUTURE_STCK 8 L2HP 227
ULT_PAST_STCK 8 L2HP 227
UNAVAILABLE_LANGUAGES (4C) MEPS 257
UNCLEAN (BIT) DSTSK 66
UNCOND 0 CCGD 31
UNEX_NOT_EXTENDED 4 DSANC 62
UNEX_OK 4 DSANC 62
UNFLATTENED 14
UNFLATTENED (BIT) BAACT 6, 7
UNFLATTENED (BIT) L2BL 208
UNFORGOTTEN_LINK_PTR (44) RMLK 310
UNFORGOTTEN_LINK_PTR (94C) RMLK 307
UNIQUE_ID (12) BAACT 6, 8
UNIQUE_ID (2) BAACT 7, 13, 17
UNIQUE_ID (22) BAACT 5, 10
UNIQUE_ID (34) BAACT 17
UNIQUE_ID (3C) BAACT 15
UNIQUE_ID (54) BAACT 10
UNIQUE_ID (6) BAACT 14
UNIQUE_ID (7A) BAACT 16
UNIQUE_ID (A) BAACT 14, 15
UNIQUE_ID (AC) BAACT 16
UNIQUE_ID (E) BAACT 12, 18
unit
  recovery manager unit of work class data, RMUW 337
  recovery manager unit of work instance, RMUW 330
UNKNOWN_EVENT 4 DMENC 53
UNLOCK_ERROR_CODE 4 DHANC 42
UNLOCK_ERROR_CODE 4 LGANC 193
UNSHUNT_ACTIVE (BIT) RMLK 312
UNSHUNT_ACTIVE (BIT) RMUW 331
UNSHUNT_DEFERRED (BIT) RMLK 312
UNSHUNT_DEFERRED (BIT) RMUW 331
UNSHUNT_Q (68) RMLK 312
UNSHUNT_Q (68) RMUW 331
UNSHUNT_REASON (0) RMUW 335
UNSHUNT_REASON_AVAIL 1 RMUW 336, 340
UNSHUNT_REASON_INDOUBT_RES 1 RMUW 336, 340
UNSHUNT_REASON_RESTART 1 RMUW 336, 340
UNSHUNT_REQUEST (0) RMUW 335
UNSHUNTED (9F0) RMLK 308
UNSHUNTED (E8) RMLK 311
UNUSED_PTR (0) DSTSK 64, 67
uow 311
  file control cfdt uow pool block, FCUPC 107
UOW_BROWSE_CHAIN_LINK (0) RMUW 334
UOW_BROWSE_CLIENT_NAME (38) RMUW 335
UOW_BROWSE_ELEMENT (0) RMUW 334
UOW_BROWSE_ENDED (34) RMUW 335
UOW_BROWSE_FILTER (35) RMUW 335
UOW_BROWSE_ITERATOR (18) RMUW 335
UOW_BROWSE_NOT_SHUNTED (36) RMUW 335
UOW_BROWSE_OWNER (30) RMUW 335
UOW_BROWSE_SHUNTED (35) RMUW 335
UOW_BROWSE_TOKEN (10) RMUW 335
UOW_BROWSE_TOKEN_SET 339
UOW_BROWSE_TOKEN_TYPE 334
UOW_BROWSE_WORK_TOKEN (37) RMUW 335
UOW_BROWSES 339
UOW_CD_EYE_CATCHER (0) RMUW 337
UOW_CHAIN 337
UOW_CHAIN_LINK (18) RMLK 311
UOW_CHAIN_LINK (18) RMUW 330
UOW_CONTEXT 311, 331
UOW_EYE_CATCHER (8) RMLK 311
UOW_EYE_CATCHER (8) RMUW 330
UOW_FACTORY (40) RMUW 337
UOW_LOG_REGISTER 338
UOW_LOGGABLE_ID 338
UOW_LOGGABLE_ID_NAME 4 RMUW 336, 340
UOW_POINTER (3C) RMLK 310
UOW_POINTER (944) RMLK 307
UOW_RO_SYNCPOINT_ORDER (E8) RMUW 338
UOW_RO_SYNCPOINT_ORDER_ARRAY (E8) RMUW 338
UOW_STATISTICS (968) RMUW 339
UOW_SURVIVED_COLD_START (BIT) RMLK 308, 310
UOW_TERMINATE_RECOVERY_NECESSARY (BIT) RMLK 307, 310
UOW_TOKEN (28) RMLK 311
UOW_TOKEN (28) RMUW 330
UOW_TOKEN_SET (100) RMUW 339
UOW_TOKEN_TYPE 340
update
  resource definition update block, RDUB 300
UPPER 1 MEPS 259
UPPERCASE_REQ (BIT) STUCB 376
UREASON (10) RMUW 335
urp
  web interface urp constants, WBUCC 424
US_ADD_LOCK_NAME 8 USANC 409
US_SCOPE_CICS 1 USANC 406
US_SCOPE_MVSIMAGE 1 USANC 406
US_SCOPE_NONE 1 USANC 406
US_SCOPE_SYSPLEX 1 USANC 406
US_STATE_INITIALIZED 1 USANC 406
US_STATE_INITIALIZING 1 USANC 406
US_STATE_QUIESCED 1 USANC 406
US_STATE_QUIESCING 1 USANC 406
US_STATE_TERMINATED 1 USANC 406
US_TXN_LOCK_NAME 8 USANC 409
USA (0) USANC 405
USA_DCEDATA_SPTOKEN (50) USANC 405
USA_DEFAULT_USER_TOKEN (78) USANC 406
USA_DEFAULT_USERID (17) USANC 405
USA_DEFAULT_USUDB_PTR (58) USANC 405
USA_DIRECTORY_NOT_FOUND_COUNT (98) USANC 406
USA_DIRECTORY_REUSE_COUNT (94) USANC 406
USA_DIRECTORY_TIMEOUT_VALUE 405
USA_DIRKEY_DIRECTORY_TOKEN (60) USANC 405
USA_ENQ_LIMIT_EXCEEDED_MSG (BIT) USANC 405
USA_EYE_CATCHER 14 USANC 409
USA_FLAGS (12) USANC 405
USA_GENERAL_SPTOKEN (30) USANC 405
USA_GENERIC_APPLID (28) USANC 405
USA_JOBSTEP_TRANS_TOKEN (70) USANC 406
USA_LAST_RESET_TIME (9C) USANC 406
USA_LOCK_TOKEN1 (80) USANC 406
USA_LOCK_TOKEN2 (84) USANC 406
USA_PREFIX (0) USANC 405
USA_PREFIX_LENGTH (0) USANC 405
USA_PREFIX_TEXT (2) USANC 405
USA_SIGNON_SCOPE (11) USANC 405
USA_TIMEOUT_EXPIRY_COUNT (90) USANC 406
USA_TIMEOUT_MEAN_REUSE_TIME (88) USANC 406
USA_TIMEOUT_REUSE_COUNT (8C) USANC 406
USA_TIMER_TOKEN (68) USANC 406
USA_US_STATE (10) USANC 405
USA_USER_TIMEOUT_QUEUE_PTR (5C) USANC 405
USA_USER_TOKEN_HWM (7C) USANC 406
USA_USERDATA_SPTOKEN (40) USANC 405
USA_USERTOKEN_DIRECTORY_TOKEN (64) USANC 405
USA_UTQE_SPTOKEN (48) USANC 405
USA_XMTRAN_SPTOKEN (38) USANC 405
USANC 405
USDK_APPLID (20) UDB 404
USDK_DIRECTORY_KEY (0) UDB 404
USDK_ENTRY_PORT (17) UDB 404
USDK_GROUPID (D) UDB 404
USDK_SCOPE_ACTIVE (A) UDB 404
USDK_USERID (0) UDB 404
USDK_UUID (28) UDB 404
USE_COUNT 208
USE_COUNT (6) DSTSK 64, 67
user
  user domain anchor block, USANC 405
  user domain statistics, USGPS 409
  user domain transaction data, USXD 410

user *(continued)*
  user domain transaction token, USXT 411
  user domain user data block, UDB 403
USER_CHAIN_HEADER (0) L2LF 234
USER_DEFAULT_LANG_PTR (11C) MEPS 257
USER_EXIT_MAP 1 MEMMS 256
USER_EXTENSION_ROOT (E0) DSANC 55
USER_MSG_MOD_PTRS (1B0) MEPS 257
USER_OPTION_FIELD (3C) SOA 371
USER_REC_TYPE 2 L2LF 237
USER_RM_START (24) L2LF 236
USER_RM_START (24) LGSF 201
USER_RM_START (4) L2LF 234
USER_TASK_ROOT (C0) DSANC 55
USER_TOKEN (34) L2CH 220
USER_TOKEN (AC) DSTSK 66
USERID 311, 331, 336
USERID (114) BAACT 11
USERID (14) BAACT 13
USERID (F4) BAACT 18
USERID_FROZEN (BIT) RMLK 312
USERID_FROZEN (BIT) RMUW 331
USERRECS (19) BAPT 23
USG_DATA_LENGTH (0) USGPS 409
USG_DIRECTORY_NOT_FOUND_COUNT (18) USGPS 409
USG_DIRECTORY_REUSE_COUNT (14) USGPS 409
USG_ID (2) USGPS 409
USG_ID_MASK 2 USGPS 410
USG_TIMEOUT_EXPIRY_COUNT (10) USGPS 409
USG_TIMEOUT_MEAN_REUSE_TIME 409
USG_TIMEOUT_REUSE_COUNT (C) USGPS 409
USG_VERSION (4) USGPS 409
USG_VERSION_MASK 1 USGPS 410
USGPS 409
USQ_DATATYPE (2C) FEP06 123
USQ_QUEUER (24) FEP06 123
USQ_RECORD (30) FEP06 123
USQ_RECORD_PTR (28) FEP06 123
USQDATA (2C) FEP06 123
USR (0) SMMCC 365
USR_CLASS (0) SMMCC 365
USR_DATA (8) SMMCC 365
USR_INITIMG (1) SMMCC 365
USR_LENGTH (2) SMMCC 365
USR_SAA (0) SMMCC 365
USR_TCAP (4) SMMCC 365
USS (BIT) STUCB 376
USS_BUFFER (0) STCB1 375
USS_CHAIN_PTR (5C) STCB1 374
USS_LOCK_TOKEN (4C) STCB1 374
USUD_ACEE_PTR (18) UDB 403
USUD_ADD_USE_COUNT (10) UDB 403
USUD_APPLID (50) UDB 404
USUD_CURRENT_GROUPID 403
USUD_DELETE_IMMEDIATE (BIT) UDB 403
USUD_ENTRY_PORT 404
USUD_GROUPID (2B) UDB 403
USUD_NATIONAL_LANGUAGE 404
USUD_OPCLASS_BYTE (5C) UDB 404
USUD_OPERATOR_CLASSES (5C) UDB 404
USUD_OPERATOR_IDENT 404
USUD_OPERATOR_PRIORITY (2A) UDB 403
USUD_SCOPE_CHECK (BIT) UDB 403
USUD_SCOPE_OBTAINED (BIT) UDB 403
USUD_SECURITY_TOKEN (8) UDB 403
USUD_TIMEOUT_INTERVAL (1C) UDB 403
USUD_TRAN_USE_COUNT (14) UDB 403
USUD_USDDB_PTR (4C) UDB 404
USUD_USER_DATA (0) UDB 403
USUD_USER_OPTIONS (1E) UDB 403
USUD_USER_TOKEN (0) UDB 403
USUD_USERID (1F) UDB 403
USUD_USERNAME 404
USUD_UTQE_TOKEN (4) UDB 403
USUD_XRF_REFLECTABLE 403
USXD 410
USXD_ACTIVE (0) USXD 410
USXD_EDF (C) USXD 410
USXD_EDF_TOKEN (18) USXD 410
USXD_FLAGS (1C) USXD 410
USXD_PRINCIPAL (4) USXD 410
USXD_PRINCIPAL_TOKEN (10) USXD 410
USXD_SESSION (8) USXD 410
USXD_SESSION_TOKEN (14) USXD 410
USXD_TRANSACTION_DATA (0) USXD 410
USXD_XS_CALLED (BIT) USXD 410
USXT 411
USXT_TRANSACTION_TOKEN (0) USXT 411
USXT_USERID_PTR (0) USXT 411
USXT_USXD_PTR (4) USXT 411
utility
  CICS affinities utility trace table, CAUTR 26
  statistics utility program anchor block, STUCB 375

V

VAL (18) USANC 405
VAL (20) UDB 403
VAL (2C) UDB 403
VAL (38) UDB 404
VAL (4) XSSS 454
VAL (68) XSSS 452
VAL (78) XSSS 452
VAL (88) XSSS 452
VARIABLE_SUBPOOL_BOUNDARY 4 TSMN 393
VBYTE (0) FEP08 130
VCA (0) TSAUX 387
VCA_CHNP (4) TSAUX 387
VCA_ECB (8) TSAUX 387
VCA_FLAGS (2) TSAUX 387
VCA_IOP (BIT) TSAUX 387
VCA_LEN (0) TSAUX 387
VCA_LOCK (BIT) TSAUX 387
VCA_RBA (C) TSAUX 387
VCA_VSWAP (10) TSAUX 387
vector
  logger reusable extended iliffe vector class, RUEI 343
VOTE (44) RMLS 319
VOTE (9EB) RMLK 308
VOTE (A4) RMLK 313
VOTE (A4) RMUW 332
VOTE (E3) RMLK 311
VOTE (FC) RMLK 314
VOTE (FC) RMUW 333
VOTER 314, 324, 333
VPLADR (BIT) CCGD 30
VPLASY (BIT) CCGD 30
VPLBWD (BIT) CCGD 30
VPLCNV (BIT) CCGD 30
VPLDIR (BIT) CCGD 30
VPLECBSW (BIT) CCGD 30
VPLGEN (BIT) CCGD 30
VPLKEY (BIT) CCGD 30
VPLKGE (BIT) CCGD 30
VPLLOC (BIT) CCGD 30
VPLLRD (BIT) CCGD 30
VPLNSP (BIT) CCGD 30
VPLOPT1 (0) CCGD 30
VPLOPT2 (0) CCGD 30
VPLSEQ (BIT) CCGD 30
VPLSKP (BIT) CCGD 30
VPLUPD (BIT) CCGD 30
VPLWAITX (BIT) CCGD 30
VSAM_ACB_A (1C) CCGD 29
VSAMCHEK 1 CCGD 31
VSAMERAS 1 CCGD 32
VSAMEREQ 1 CCGD 32
VSAMGET 1 CCGD 31
VSAMPNT 1 CCGD 32
VSAMPUT 1 CCGD 31
VTAM
  VTAM acb work area, FEP03 115
  VTAM receive request block, FEP15 139
  VTAM requests block, FEP16 140

W

wait
  domain manager wait queue element, DMCB3 50
  file control cfdt pool wait element, FCPWC 102
  temporary storage wait queue class, TSWQ 402
WAIT 0 CCGD 31
WAIT_END 0 CCGD 31
WAIT_FINISH 59
WAIT_QUEUE (0) DMCB3 50

WAIT_RESOURCE_TYPE_WRITE 8 L2HS 231
WAIT_START (58) DSANC 59
WAIT_TOKEN (6C) DSTSK 66
WAIT_TYPE (74) DSTSK 66
WAIT_WRITE_ISSUED 1 L2SR 251
WAIT_XC (BIT) CCGD 30
waiter
  enqueue domain browse waiter extension, NQWX 282
WAKE_UP_ECB (28) DSANC 59
WARM 1 PAA 284
WARM_KP_WAITING_FOR_AKP_END (1E) RMSL 327, 329
WB_GENERAL 8 WBANC 413
WB_LOCK_NAME 8 WBANC 413
WB_STATE_INITIALISED 1 WBANC 413
WB_STATE_INITIALISING 1 WBANC 413
WB_STATE_QUIESCED 1 WBANC 413
WB_STATE_QUIESCING 1 WBANC 413
WB_STATE_TERMINATED 1 WBANC 413
WBA (0) WBANC 412
WBA_3270_ANCHOR (34) WBANC 412
WBA_BUFFER_TOKEN (20) WBANC 412
WBA_CCNV_LOAD_OK (BIT) WBANC 412
WBA_CODEPAGE_NAME (40) WBANC 412
WBA_CODEPAGE_NUMBER 412
WBA_COLD_START (BIT) WBANC 412
WBA_CONVTABL (11) WBANC 412
WBA_END (48) WBANC 412
WBA_EYE_CATCHER 14 WBANC 413
WBA_FLAGS (29) WBANC 412
WBA_GENERAL_SPTOKEN (18) WBANC 412
WBA_LENGTH (0) WBANC 412
WBA_LOCK_TOKEN (10) WBANC 412
WBA_PREFIX (0) WBANC 412
WBA_PREFIX_TEXT (2) WBANC 412
WBA_STARTUP_FLAGS (10) WBANC 412
WBA_STATE_ANCHOR_PTR (14) WBANC 412
WBA_TTABL (0) WBANC 412
WBA_TTABL_EYECATCH (2) WBANC 412
WBA_TTABL_HDR (0) WBANC 412
WBA_TTABL_LEN (0) WBANC 412
WBA_UNESCAPE_CODEPAGE_PTR (38) WBANC 412
WBA_UNESCAPE_TABLE_INITIALIZED 412
WBA_WARM_START (BIT) WBANC 412
WBA_WB_STATE (28) WBANC 412
WBA_WBUD_USED (BIT) WBANC 412
WBA_WEBREQUEST_CLASSP (30) WBANC 412
WBA1_CLIENT_ADDRESS (10) WBA1C 414
WBA1_CLIENT_ADDRESS_LENGTH (23) WBA1C 414
WBA1_CLIENT_ADDRESS_STRING (14) WBA1C 414
WBA1_CONVERTER_PROGRAM_NAME (8) WBA1C 414
WBA1_DATA (66) WBA1C 415
WBA1_DATA_OFFSET (28) WBA1C 414
WBA1_DATA_PTR 414
WBA1_EYECATCHER (0) WBA1C 414
WBA1_EYECATCHER_BLIO 8 WBA1C 415
WBA1_EYECATCHER_BLIP 8 WBA1C 415
WBA1_HEADER_LENGTH (46) WBA1C 415
WBA1_HEADER_OFFSET (38) WBA1C 415
WBA1_HTTP_VERSION_LENGTH (42) WBA1C 415
WBA1_HTTP_VERSION_OFFSET (30) WBA1C 415
WBA1_INPUT_DATA_LENGTH 415
WBA1_METHOD_LENGTH (40) WBA1C 415
WBA1_METHOD_OFFSET (2C) WBA1C 414
WBA1_OUTDATA_OFFSET (60) WBA1C 415
WBA1_OUTDATA_PTR (60) WBA1C 415
WBA1_PARMS (0) WBA1C 414
WBA1_PARMS_PLIST (0) WBA1C 414
WBA1_RESOURCE_LENGTH (44) WBA1C 415
WBA1_RESOURCE_OFFSET (34) WBA1C 415
WBA1_RESPONSE (64) WBA1C 415
WBA1_SERVER_PROGRAM_NAME (50) WBA1C 415
WBA1_USER_DATA_LENGTH (48) WBA1C 415
WBA1_USER_DATA_OFFSET (3C) WBA1C 415
WBA1_USER_TOKEN (58) WBA1C 415
 WBA1C 413
WBAB_3270_ENVIRONMENT_TOKEN 411
WBAB_ANCHOR_LENGTH (0) WBABC 411
WBAB_BUFFER_TOKEN (38) WBABC 411
WBAB_DFHWBST_ENTRY_POINT 411
WBAB_DFHWBTC_ENTRY_POINT (18) WBABC 411
WBAB_EYECATCHER (2) WBABC 411
WBAB_MDT_TOKEN (48) WBABC 411
WBAB_OPENEDITION_UID (40) WBABC 411
WBAB_PREFIX (0) WBABC 411
WBAB_STATE_ANCHOR_PTR 411
WBAB_STATE_TOKEN (30) WBABC 411
WBAB_TEMPLATE_ANCHOR_PTR (24) WBABC 411
WBAB_UNESCAPE_CODEPAGE_PTR (44) WBABC 411
WBAB_WEB_ANCHOR_BLOCK (0) WBABC 411
 WBABC 411
 WBANC 412
WBBL_ARROW (2) WBBLC 417
WBBL_BLOCK_NAME (8) WBBLC 417
WBBL_CLIENT_ADDRESS (1C) WBBLC 417
WBBL_CLIENT_ADDRESS_LENGTH (20) WBBLC 417
WBBL_CLIENT_ADDRESS_STRING (21) WBBLC 417
WBBL_CLIENT_CERTIFICATE (90) WBBLC 418
WBBL_CLIENT_CERTIFICATE_LENGTH (8C) WBBLC 418
WBBL_CLIENT_CERTIFICATE_OFFSET (88) WBBLC 418
WBBL_COMPID (6) WBBLC 417
WBBL_CONVERTER_PROGRAM_NAME (30) WBBLC 418
WBBL_CURRENT_VERSION 4 WBBLC 419
WBBL_DATA (90) WBBLC 418
WBBL_DFH (3) WBBLC 417
WBBL_EYECATCHER (2) WBBLC 417
WBBL_HEADER_LENGTH (7C) WBBLC 418
WBBL_HEADER_OFFSET (78) WBBLC 418
WBBL_HTTP_VERSION_LENGTH (6C) WBBLC 418
WBBL_HTTP_VERSION_OFFSET (68) WBBLC 418
WBBL_INDATA_LENGTH (54) WBBLC 418
WBBL_INDATA_OFFSET (50) WBBLC 418
WBBL_INDATA_PTR (50) WBBLC 418
WBBL_LENGTH (0) WBBLC 417
WBBL_METHOD_LENGTH (64) WBBLC 418
WBBL_METHOD_OFFSET (60) WBBLC 418
WBBL_MODE (11) WBBLC 417
WBBL_MODE_OFFSET 1 WBBLC 419
WBBL_MODE_POINTER 1 WBBLC 419
WBBL_OUTDATA_LENGTH (5C) WBBLC 418
WBBL_OUTDATA_OFFSET (58) WBBLC 418
WBBL_OUTDATA_PTR (58) WBBLC 418
WBBL_PARMS (0) WBBLC 417
WBBL_PARMS_PLIST (0) WBBLC 417
WBBL_PREFIX (0) WBBLC 417
WBBL_PROLOG (18) WBBLC 417
WBBL_PROLOG_SIZE (14) WBBLC 417
WBBL_RESOURCE_LENGTH (74) WBBLC 418
WBBL_RESOURCE_OFFSET (70) WBBLC 418
WBBL_RESPONSE (18) WBBLC 417
WBBL_SERVER_ADDRESS (48) WBBLC 418
WBBL_SERVER_PORTNUMBER (4C) WBBLC 418
WBBL_SERVER_PROGRAM_NAME (38) WBBLC 418
WBBL_SSL_KEYSIZE (4E) WBBLC 418
WBBL_STATUS (10) WBBLC 417
WBBL_STATUS_SIZE (10) WBBLC 417
WBBL_USER_DATA_LENGTH (84) WBBLC 418
WBBL_USER_DATA_OFFSET (80) WBBLC 418
WBBL_USER_TOKEN (40) WBBLC 418
WBBL_VECTOR (50) WBBLC 418
WBBL_VECTOR_SIZE (16) WBBLC 417
WBBL_VERSION (12) WBBLC 417
WBBL_VERSION_CTS130 4 WBBLC 419
 WBBLC 416
WBEP_ABEND_CODE 421
WBEP_ANALYZER_REASON (70) WBEPC 421
WBEP_ANALYZER_RESPONSE (6C) WBEPC 421
WBEP_CLIENT_ADDRESS (29) WBEPC 421
WBEP_CLIENT_ADDRESS_LEN (28) WBEPC 421
WBEP_CONVERTER_PROGRAM (50) WBEPC 421
WBEP_CONVERTER_REASON (78) WBEPC 421
WBEP_CONVERTER_RESPONSE (74) WBEPC 421
WBEP_DATA (C) WBEPC 421
WBEP_ERROR_CODE (C) WBEPC 421
WBEP_EYECATCHER (2) WBEPC 421
WBEP_FAILING_PROGRAM (60) WBEPC 421
WBEP_HTTP_RESPONSE_CODE (68) WBEPC 421
WBEP_LENGTH (0) WBEPC 421
WBEP_MESSAGE_LEN (1C) WBEPC 421
WBEP_MESSAGE_NUMBER (14) WBEPC 421
WBEP_MESSAGE_PTR (18) WBEPC 421
WBEP_PREFIX (0) WBEPC 421
WBEP_RESPONSE_LEN (24) WBEPC 421
WBEP_RESPONSE_PTR (20) WBEPC 421
WBEP_SERVER_ADDRESS (39) WBEPC 421
WBEP_SERVER_ADDRESS_LEN (38) WBEPC 421
WBEP_TARGET_PROGRAM (58) WBEPC 421

WBEP_TCPIPSERVICE_NAME (48) WBEPC 421
WBEP_VERSION (A) WBEPC 421
WBEPC 419
WBSTA_ANCHOR_BLOCK (0) WBSTC 423
WBSTA_ANCHOR_PREFIX (0) WBSTC 423
WBSTA_ANCHOR_PREFIX_LEN (0) WBSTC 423
WBSTA_ANCHOR_PREFIX_TEXT (2) WBSTC 423
WBSTA_DIRECTORY_TOKEN (14) WBSTC 423
WBSTA_GARBAGE_INTERVAL (10) WBSTC 423
WBSTA_LOCK_TOKEN (18) WBSTC 423
WBSTA_TERMINAL_TIMEOUT (20) WBSTC 423
WBSTA_WAKEUP_TIME (1C) WBSTC 423
WBSTC 422
WBSTH_BROKEN 1 WBSTC 423
WBSTH_INITIALIZED 1 WBSTC 423
WBSTH_M_C_CODE 422
WBSTH_MADE 1 WBSTC 423
WBSTH_MASTER_CUOWID (18) WBSTC 422
WBSTH_MASTER_ECB (20) WBSTC 422
WBSTH_MASTER_TASKID (14) WBSTC 422
WBSTH_NOT_INITIALIZED 1 WBSTC 423
WBSTH_PARTNERSHIP_STATUS (10) WBSTC 422
WBSTH_PREFIX (0) WBSTC 422
WBSTH_PREFIX_LENGTH (0) WBSTC 422
WBSTH_PREFIX_TEXT (2) WBSTC 422
WBSTH_S_C_CODE 422
WBSTH_SLAVE_CUOWID (28) WBSTC 422
WBSTH_SLAVE_ECB (30) WBSTC 422
WBSTH_SLAVE_TASKID (24) WBSTC 422
WBSTH_STATE_BLOCK (0) WBSTC 422
WBSTH_TERMINATED 1 WBSTC 423
WBSTH_TIMESTAMP (34) WBSTC 422
WBSTH_USER_DATA (38) WBSTC 422
WBSTU_3270_PAGE_TOKEN (28) WBSTC 422
WBSTU_AID (51) WBSTC 422
WBSTU_BMS_PAGE_TOKEN (20) WBSTC 422
WBSTU_CONVERSATION_TYPE (50) WBSTC 422
WBSTU_CURSOR (52) WBSTC 422
WBSTU_DATA_TYPE (BIT) WBSTC 423
WBSTU_EXPORTED_DOCUMENT (48) WBSTC 422
WBSTU_EXPORTED_DOCUMENT_LEN (4C) WBSTC 422
WBSTU_EXPORTED_DOCUMENT_PTR (48) WBSTC 422
WBSTU_FACILITY_TOKEN (0) WBSTC 422
WBSTU_INITIAL_RECEIVE (BIT) WBSTC 423
WBSTU_INPUT_DATA_LENGTH (44) WBSTC 422
WBSTU_INPUT_DATA_PTR (40) WBSTC 422
WBSTU_LAST_SEND_WSF_QUERY (BIT) WBSTC 423
WBSTU_MAP_CONVERSATION 1 WBSTC 423
WBSTU_MDT_TABLE_PTR (2C) WBSTC 422
WBSTU_NEW_CONVERSATION 1 WBSTC 423
WBSTU_NEXT_TRANSACTION_ID (C) WBSTC 422
WBSTU_OUTPUT_DATA_LENGTH (34) WBSTC 422
WBSTU_OUTPUT_DATA_PTR (30) WBSTC 422
WBSTU_OUTPUT_LENGTH_REMAINING (3C) WBSTC 422
WBSTU_OUTPUT_OFFSET (38) WBSTC 422
WBSTU_PSEUDO_CONVERSATION (BIT) WBSTC 423
WBSTU_STATE_DATA (0) WBSTC 422
WBSTU_TARGET_ABEND_CODE (14) WBSTC 422
WBSTU_TARGET_TRANSACTION_ID (8) WBSTC 422
WBSTU_TC_CONVERSATION 1 WBSTC 423
WBSTU_TCPIPSERVICE (18) WBSTC 422
WBSTU_TERMID (10) WBSTC 422
WBSTU_TEXT_CONVERSATION 1 WBSTC 423
WBSTU_URL (58) WBSTC 423
WBSTU_URL_LENGTH (57) WBSTC 423
WBSTU_USER_STATE (54) WBSTC 423
WBUCC 424
WCIB (0) TSAUX 387
web
  web anchor block, WBABC 411
  web business logic compatibility interface, WBA1C 413
  web business logic interface parameters, WBBLC 416
  web domain anchor block, WBANC 412
  web error program parms, WBEPC 419
  web interface urp constants, WBUCC 424
  web request block class, WRB 427
  web state manager data, WBSTC 422
WEBREQ (0) WRB 427
WEBREQUEST_ANCHOR 8 WBANC 413
WEIGHTED_AVERAGE_PERIOD 4 SMDCC 363
WILDCHAR 1 TSMN 391
WL (608) DSANC 58
WL_AVERAGE 58
WL_AVERAGE_DURATION (608) DSANC 58
WL_DURATION (624) DSANC 58
WL_FIRST (61C) DSANC 58
WL_LAST (620) DSANC 58
WL_N (614) DSANC 58
WL_OLDEST (618) DSANC 58
WL_SUM 58
work
  cics/db2 global work area, D2GWA 92
  file browse work area for data tables, FBWAC 99
  language interface work area, APLI 3
  recovery manager unit of work class data, RMUW 337
  recovery manager unit of work instance, RMUW 330
  VTAM acb work area, FEP03 115
  work queue element, FEP14 138
WQ_ARROW (2) DMCB3 50
WQ_BLOCK_NAME (8) DMCB3 50
WQ_CALLER_DOMAIN (18) DMCB3 50
WQ_DFH (3) DMCB3 50
WQ_DOMAIN_TOKEN (1C) DMCB3 50
WQ_DOMID (6) DMCB3 50
WQ_HEAD (77C) DMCB1 47
WQ_HEAD_BLOCK_NAME 8 DMCB3 50
WQ_LENGTH (0) DMCB3 50
WQ_NEXT (10) DMCB3 50
WQ_PHASE (20) DMCB3 50
WQ_PREFIX (0) DMCB3 50
WQ_PREV (14) DMCB3 50
WQ_SUSP_TOKEN 50
WRA 427
WRA_ARROW (2) WRB 427
WRA_BLOCK_NAME (8) WRB 427
WRA_DFH (3) WRB 427
WRA_DOMID (6) WRB 427
WRA_LENGTH (0) WRB 427
WRA_PREFIX (0) WRB 427
WRA_WRB_FIRST 427
WRA_WRB_LAST (24) WRB 427
WRA_WRB_SPTOKEN (10) WRB 427
WRA_WRBHEAD 427
WRA_WRBR_FIRST (28) WRB 427
WRA_WRBR_LAST (2C) WRB 427
WRA_WRBR_SPTOKEN (18) WRB 427
WRA_WRBRHEAD (28) WRB 427
WRB 427
WRB (0) WRB 427
WRB_ABEND_CODE (180) WRB 429
WRB_ANALYZER_NAME (E0) WRB 429
WRB_ANALYZER_REASON (EC) WRB 429
WRB_ANALYZER_RESPONSE (E8) WRB 429
WRB_BYTES_RECEIVED (B0) WRB 429
WRB_CERT_REPOSITORY_TOKEN (118) WRB 429
WRB_CHAR_CLIENT_ADDRESS (59) WRB 428
WRB_CHAR_CLIENT_ADDRESS_AREA (58) WRB 428
WRB_CHAR_CLIENT_ADDRESS_LEN (58) WRB 428
WRB_CHAR_SERVER_ADDRESS (69) WRB 428
WRB_CHAR_SERVER_ADDRESS_AREA (68) WRB 428
WRB_CHAR_SERVER_ADDRESS_LEN (68) WRB 428
WRB_CLIENT_ADDRESS (50) WRB 428
WRB_CLIENT_CODEPAGE (120) WRB 429
WRB_COMMON (78) WRB 428
WRB_CONNECTION_PERSISTENT (BIT) WRB 428
WRB_CONTENT_LENGTH (B4) WRB 429
WRB_CONTENT_LENGTH_FOUND (BIT) WRB 428
WRB_CONTENT_LENGTH_SENT (BIT) WRB 428
WRB_CONVERTER_PROGRAM_NAME (40) WRB 428
WRB_CONVERTER_REASON (F4) WRB 429
WRB_CONVERTER_RESPONSE (F0) WRB 429
WRB_CURRENT_PTR (B8) WRB 429
WRB_DFHCNV_KEY (C4) WRB 429
WRB_ERROR_CODE (184) WRB 429
WRB_EXEC_CICS_WEB_SEND (BIT) WRB 428
WRB_EYECATCHER (2) WRB 427
WRB_FAILING_PROGRAM (170) WRB 429
WRB_FIRST_LINE_COMPLETE (BIT) WRB 428
WRB_FIRST_LINE_LENGTH (DC) WRB 429
WRB_FIRST_RECV_IN_REQUEST (BIT) WRB 428
WRB_FLAGS1 (18) WRB 427
WRB_FLAGS2 (19) WRB 428
WRB_GREATER_THAN_32K (BIT) WRB 427
WRB_HEADER_BROWSE_OFFSET (FC) WRB 429
WRB_HEADER_BROWSE_TOKEN (F8) WRB 429
WRB_HEADER_LENGTH (94) WRB 428

WRB_HEADER_NUMBER (AC) WRB 429
WRB_HEADER_OFFSET (90) WRB 428
WRB_HEADERS_RECEIVED (BIT) WRB 428
WRB_HTTP_VERSION_LENGTH (8C) WRB 428
WRB_HTTP_VERSION_OFFSET (88) WRB 428
WRB_INITIAL_BUFFER (BIT) WRB 428
WRB_INITIAL_STRING (178) WRB 429
WRB_INPUT_DATA_LENGTH 429
WRB_KEEP_ALIVE_SENT (BIT) WRB 428
WRB_KEYSIZE 428
WRB_LENGTH (0) WRB 427
WRB_METHOD_LENGTH (7C) WRB 428
WRB_METHOD_OFFSET (78) WRB 428
WRB_NEW_SEND_DOCTOKEN (158) WRB 429
WRB_NEXT (10) WRB 427
WRB_OUTDATA_LENGTH (C0) WRB 429
WRB_OUTDATA_PTR (BC) WRB 429
WRB_OVERLEN_DATA_PTR (154) WRB 429
WRB_PREFIX (0) WRB 427
WRB_PREV (14) WRB 427
WRB_RECEIVE_BUFFER_OFFSET (A8) WRB 429
WRB_RECEIVE_COMPLETE (BIT) WRB 428
WRB_RECEIVE_DATA_PTR (150) WRB 429
WRB_REMAINING_BUFFER_LEN (34) WRB 428
WRB_REPOSITORY_HEADER (110) WRB 429
WRB_REPOSITORY_STCK (D8) WRB 429
WRB_REPOSITORY_TOKEN (108) WRB 429
WRB_REQUEST_TYPE (24) WRB 428
WRB_RESOURCE_LENGTH (84) WRB 428
WRB_RESOURCE_OFFSET (80) WRB 428
WRB_RESPONSE_HEADER_LEN (104) WRB 429
WRB_RESPONSE_LINE_LENGTH (168) WRB 429
WRB_ROUNDED_UP_LENGTH 4 WRB 430
WRB_SEND_BODY_LENGTH (16C) WRB 429
WRB_SEND_DOCUMENT (BIT) WRB 428
WRB_SEND_RESPONSE_FAILED (BIT) WRB 428
WRB_SERVER_ADDRESS (54) WRB 428
WRB_SERVER_DATA_PTR (30) WRB 428
WRB_SERVER_PORTNUMBER (116) WRB 429
WRB_SERVER_PROGRAM_NAME (38) WRB 428
WRB_SERVER_PROTOCOL (CC) WRB 429
WRB_SESSION_TOKEN 428
WRB_SESSION_TOKEN_PART1 (1C) WRB 428
WRB_SESSION_TOKEN_PART2 (20) WRB 428
WRB_SHARED_TS_REPOSITORY (BIT) WRB 428
WRB_TASK_NUM (D4) WRB 429
WRB_TCPIPSERVICE (148) WRB 429
WRB_TIDYUP_COMPLETE (BIT) WRB 428
WRB_USER_DATA_CURSOR (100) WRB 429
WRB_USER_DATA_ESCAPED (BIT) WRB 428
WRB_USER_DATA_LENGTH (9C) WRB 428
WRB_USER_DATA_OFFSET (98) WRB 428
WRB_USER_NUMBER (AE) WRB 429
WRB_USER_TOKEN (48) WRB 428
WRB_USERID (28) WRB 428
WRBR 429
WRBR_CHANGE_COUNT (1C) WRB 429
WRBR_NEXT (0) WRB 429
WRBR_PREV (4) WRB 429
WRBR_TOKEN (18) WRB 429
WRBR_TRANID (8) WRB 429
WRBR_TRANNUM (C) WRB 429
WRBR_TRANTOKEN (10) WRB 429
WRBR_WRBP (20) WRB 429
WRITE_ANSA (168) L2BS 216
WRITE_ANSA (168) L2SR 248
WRITE_ANSA (78) L2HS 230
WRITE_ECB (164) L2BS 216
WRITE_ECB (164) L2SR 248
WRITE_ECB (74) L2HS 230
WRITE_LIST_ADDR (28) SOA 371
WRITE_LIST_LENGTH (24) SOA 371
WRITE_PARMS (814) STUCB 375
WRITEABLE (BIT) L2BL 208
WRITING_REPORT_SUMM (BIT) STUCB 376
WRITING_SUMMARY (BIT) STUCB 376
WRQ_ANALYZER_DATALENG_ERROR 4 WRB 430
WRQ_ANALYZER_ERROR 4 WRB 430
WRQ_ANALYZER_LINK_ERROR 4 WRB 430
WRQ_BAD_PREVIOUS_SEND 4 WRB 430
WRQ_CODEPAGE_NOT_FOUND 4 WRB 430
WRQ_CONNECTION_CLOSED 4 WRB 430
WRQ_DISASTER 4 WRB 430
WRQ_DOCUMENT_NOT_FOUND 4 WRB 430
WRQ_HDR_BROWSE_ACTIVE 4 WRB 430
WRQ_HDR_BROWSE_END 4 WRB 430
WRQ_HDR_BROWSE_NOT_ACTIVE 4 WRB 430
WRQ_HDR_LENGTH_ERROR 4 WRB 430
WRQ_HDR_NAME_LENGTH_ERROR 4 WRB 430
WRQ_HDR_NOT_FOUND 4 WRB 430
WRQ_HDR_VALUE_LENGTH_ERROR 4 WRB 430
WRQ_INVALID_HEADER 4 WRB 430
WRQ_INVALID_REQUEST_FORMAT 4 WRB 430
WRQ_NO_ANALYZER 4 WRB 430
WRQ_NO_PREVIOUS_SEND 4 WRB 430
WRQ_NOT_HTTP_REQUEST 4 WRB 430
WRQ_NOT_WEB_REQUEST 4 WRB 430
WRQ_OK 4 WRB 430
WRQ_PURGED 4 WRB 430
WRQ_REPOSITORY_IO_ERROR 4 WRB 430
WRQ_RESPONSE (0) WRB 430
WRQ_SOCKETS_CLOSE_ERROR 4 WRB 430
WRQ_SOCKETS_RECEIVE_ERROR 4 WRB 430
WRQ_SOCKETS_SEND_ERROR 4 WRB 430
WRQ_SOIS_INQUIRE_FAILED 4 WRB 430
WRQ_STORAGE_ERROR 4 WRB 430
WRQ_WBQM_GET_BODY_OUT_FAILED 4 WRB 430
WRQ_WBQM_GET_HEADER_OUT_FAILED 4 WRB 430
WRQ_WBQM_GET_REPTOKEN_ERR 4 WRB 430
WRQ_WBQM_GET_RESPLINE_FAILED 4 WRB 430
WRQ_WBQM_PUT_HEADER_FAILED 4 WRB 430
WRQ_WBQM_PUT_USER_FAILED 4 WRB 430

X

XA 2 CCGD 31
XBYTE (0) FEP08 130
XCCBC 431
XCDMP_NO_SVCNUM 2 XCCBC 434
XCEIP_CANNOT_CALL_XCDMP 2 XCCBC 434
XCEIP_ESTAE_SETUP 2 XCCBC 434
XCEIP_NO_RETCODE_AREA 2 XCCBC 434
XCEIP_UNSUPPORTED_COMMAND 2 XCCBC 434
XCG_CURRENT_XCP (94) XCCBC 432
XCG_CURRENT_XCU (90) XCCBC 432
XCG_DMP_ADDR (24) XCCBC 431
XCG_DUMP_ERROR_DATA (84) XCCBC 432
XCG_DUMP_FLAGS (81) XCCBC 432
XCG_DUMP_NUM (64) XCCBC 432
XCG_DUMP_STR (78) XCCBC 432
XCG_DUMP_TITLE_LEN (6C) XCCBC 432
XCG_DUMP_TITLE_PTR (68) XCCBC 432
XCG_DUMPCODE (70) XCCBC 432
XCG_EIP_ADDR (18) XCCBC 431
XCG_EIP_WS (3C) XCCBC 431
XCG_EIP_WS_LEN (44) XCCBC 431
XCG_EYE (2) XCCBC 431
XCG_GTF_STARTED (BIT) XCCBC 431
XCG_INT_MSG 432
XCG_INT_MSG_0 (AA) XCCBC 432
XCG_INT_MSG_LEN (A8) XCCBC 432
XCG_INT_MSG_TEXT (AC) XCCBC 432
XCG_IRP_CHK_FLAGS (A4) XCCBC 432
XCG_IRP_LEVEL (A0) XCCBC 432
XCG_JOBNAME (12E) XCCBC 432
XCG_JOBNAME_LEN (12C) XCCBC 432
XCG_LENGTH (0) XCCBC 431
XCG_LEVEL_CHECKED (BIT) XCCBC 432
XCG_LEVEL_OK (BIT) XCCBC 432
XCG_MSG_ADDR (30) XCCBC 431
XCG_MSG_FLAGS 432
XCG_MSG_UPPERCASE (BIT) XCCBC 432
XCG_MTAB_ADDR (34) XCCBC 431
XCG_PREFIX (0) XCCBC 431
XCG_PRH_ADDR (10) XCCBC 431
XCG_PRH_WS (38) XCCBC 431
XCG_PRH_WS_LEN (40) XCCBC 431
XCG_PROGRAM (4C) XCCBC 431
XCG_RETRY_TIME 432
XCG_SDUMP_IN_PROGRESS (BIT) XCCBC 432
XCG_SECURITY_FLAGS 432
XCG_SURROGATE_CHK (BIT) XCCBC 432
XCG_SVC_INS (98) XCCBC 432
XCG_TCB (88) XCCBC 432
XCG_TIMEOUT 432

XCG_TRA_ADDR (2C) XCCBC 431
XCG_TRACE_ANCHOR (58) XCCBC 431
XCG_TRACE_CONFDATA (BIT) XCCBC 432
XCG_TRACE_FLAGS 431
XCG_TRACE_LVL (60) XCCBC 431
XCG_TRACE_TABLE_SIZE (5C) XCCBC 431
XCG_TRAP_ACTIVE (BIT) XCCBC 432
XCG_TRAP_WA_PTR (54) XCCBC 431
XCG_TRI_ADDR (20) XCCBC 431
XCG_TRP_ADDR (1C) XCCBC 431
XCG_URM_ADDR (28) XCCBC 431
XCG_URM_ANCHOR (48) XCCBC 431
XCG_WTO_PARMS (128) XCCBC 432
XCG_XCUSER_PTR (8C) XCCBC 432
XCG_XFQ_ADDR (14) XCCBC 431
XCGLOBAL (0) XCCBC 431
XCGLOBAL_EYECATCHER 14 XCCBC 434
XCP_ALLOC_OPTS (2A) XCCBC 433
XCP_ARG_0 (178) XCCBC 433
XCP_ARG_1 (17C) XCCBC 433
XCP_ARG_2 (180) XCCBC 433
XCP_ARG_3 (184) XCCBC 433
XCP_ARG_4 (188) XCCBC 433
XCP_ARG_5 (18C) XCCBC 433
XCP_ARG_6 (190) XCCBC 433
XCP_ARG_7 (194) XCCBC 433
XCP_BIND (C8) XCCBC 433
XCP_CICS_NAME (14) XCCBC 433
XCP_CONV_STATE 433
XCP_DATA_1 (158) XCCBC 433
XCP_DATA_2 (160) XCCBC 433
XCP_DATA_3 (168) XCCBC 433
XCP_DATA_4 (170) XCCBC 433
XCP_EID (198) XCCBC 433
XCP_EYE (2) XCCBC 433
XCP_FLAGS 433
XCP_IRCLS (3C) XCCBC 433
XCP_IRCSB (40) XCCBC 433
XCP_IRP_DLENGTH (34) XCCBC 433
XCP_IRP_IO_LEN (30) XCCBC 433
XCP_IRP_IOAREA 433
XCP_LEN_1 (15C) XCCBC 433
XCP_LEN_2 (164) XCCBC 433
XCP_LEN_3 (16C) XCCBC 433
XCP_LEN_4 (174) XCCBC 433
XCP_LENGTH (0) XCCBC 433
XCP_LOGON_NAME (1C) XCCBC 433
XCP_LSLCB (14C) XCCBC 433
XCP_LUSERID (148) XCCBC 433
XCP_NEXT_XCP (10) XCCBC 433
XCP_OPEN_STATUS (28) XCCBC 433
XCP_PIPE_STATUS (28) XCCBC 433
XCP_PREFIX (0) XCCBC 433
XCP_RH_I1 (1B4) XCCBC 433
XCP_RH_I2 (1B5) XCCBC 433
XCP_RH_I3 (1B6) XCCBC 433
XCP_RH_INPUT (1B4) XCCBC 433
XCP_RH_O1 (1B7) XCCBC 433
XCP_RH_O2 (1B8) XCCBC 433
XCP_RH_O3 (1B9) XCCBC 433
XCP_RH_OUTPUT (1B7) XCCBC 433
XCP_SCCB (154) XCCBC 433
XCP_THRDID (150) XCCBC 433
XCP_UU_FMH (68) XCCBC 433
XCP_XCUSER_PTR (24) XCCBC 433
XCP_XFRASTG1 (38) XCCBC 433
XCPIPE (0) XCCBC 433
XCPIPE_EYECATCHER 14 XCCBC 434
XCPRH_CANNOT_CALL_XCDMP 2 XCCBC 434
XCPRH_ESTAE_SETUP_FAILURE 2 XCCBC 434
XCPRH_INCORRECT_SVC_LEVEL 2 XCCBC 434
XCPRH_SSI_VERIFY_FAIL 2 XCCBC 434
XCPRH_SVC_CALL_FAIL 2 XCCBC 434
XCPRH_VERIFY_GM_ERROR 2 XCCBC 434
XCPRH_WS_GM_FAILURE 2 XCCBC 434
XCPRH_XCGLOBAL_GM_ERROR 2 XCCBC 434
XCPRH_XCUSER_GM_FAILURE 2 XCCBC 434
XCSTB_CALLED_IN_AMODE24 2 XCCBC 434
XCTRI_DISASTER 1 XCCBC 434
XCTRI_FUNCTION (0) XCCBC 433
XCTRI_INITIALISE 1 XCCBC 434
XCTRI_OK 1 XCCBC 434
XCTRI_PLIST (0) XCCBC 433
XCTRI_RECOVERY 1 XCCBC 434
XCTRI_RESPONSE (1) XCCBC 433
XCTRI_TERMINATE 1 XCCBC 434
XCTRI_WS 433
XCTRI_XCG_PTR (8) XCCBC 433
XCU_APPL_NAME (10) XCCBC 432
XCU_EYE (2) XCCBC 432
XCU_FMH07_MSG (28) XCCBC 432
XCU_LENGTH (0) XCCBC 432
XCU_MSG_0 (2A) XCCBC 432
XCU_MSG_LEN (28) XCCBC 432
XCU_MSG_TEXT (2C) XCCBC 432
XCU_NEXT_XCU (1C) XCCBC 432
XCU_PIPE_PTR (20) XCCBC 432
XCU_PREFIX (0) XCCBC 432
XCU_WS_ADDR (24) XCCBC 432
XCU_XCG_PTR (18) XCCBC 432
XCUSER (0) XCCBC 432
XCUSER_EYECATCHER 14 XCCBC 434
XM_STATE_CATALOG_RECORD (0) XMCAT 438
XM_TCLASS (0) XMCLC 439
XM_TXN (0) XMXNC 445
XM_TXN_ABEND_CODE (74) XMXNC 446
XM_TXN_ABEND_IN_PROGRESS (78) XMXNC 446
XM_TXN_AP_TOKEN 446
XM_TXN_APPC_SESSION 1 XMXNC 447
XM_TXN_ATTACH_MESSAGE (16) XMXNC 445
XM_TXN_ATTACH_PARMS_ADDR (24) XMXNC 445
XM_TXN_ATTACH_PARMS_LENGTH (28) XMXNC 446
XM_TXN_ATTACH_TIME (50) XMXNC 446
XM_TXN_BR_TOKEN (F8) XMXNC 447
XM_TXN_BRIDGE 1 XMXNC 447
XM_TXN_BROWSE_COUNT (14) XMXNC 445
XM_TXN_CREATED_BY_ATTACH (BIT) XMXNC 445
XM_TXN_DEFERRED_ABEND (114) XMXNC 447
XM_TXN_DEFERRED_ABEND_SET (BIT) XMXNC 445
XM_TXN_DEFERRED_ABEND_TXN_DUMP (BIT) XMXNC 445
XM_TXN_DEFERRED_MESSAGE_SET (BIT) XMXNC 445
XM_TXN_DS_ATTACHED 1 XMXNC 447
XM_TXN_DS_TASK_TOKEN 446
XM_TXN_EXTERNAL_UOW_ID (118) XMXNC 447
XM_TXN_EYECATCHER (2) XMXNC 445
XM_TXN_FACILITY_TOKEN (18) XMXNC 445
XM_TXN_FACILITY_TYPE (10) XMXNC 445
XM_TXN_FLAGS (13) XMXNC 445
XM_TXN_FLAGS2 (17) XMXNC 445
XM_TXN_FORCE_PURGE_ISSUED (BIT) XMXNC 445
XM_TXN_GROUP_ID_INHERITED (BIT) XMXNC 445
XM_TXN_IIOP 1 XMXNC 447
XM_TXN_INFINITE_WAIT (BIT) XMXNC 445
XM_TXN_INIT_PURGE_PROTECT (BIT) XMXNC 445
XM_TXN_INSUFF_STG_MSG_ISSUED (BIT) XMXNC 445
XM_TXN_LENGTH 445
XM_TXN_LG_TOKEN (E0) XMXNC 447
XM_TXN_LU61_SESSION 1 XMXNC 447
XM_TXN_MN_TOKEN (A0) XMXNC 446
XM_TXN_MRO_SESSION 1 XMXNC 447
XM_TXN_MXT_SCHEDULED 1 XMXNC 447
XM_TXN_MXT_WAIT_START (60) XMXNC 446
XM_TXN_MXT_WAIT_TIME (60) XMXNC 446
XM_TXN_NEXT_TCLASS_WAITER (108) XMXNC 447
XM_TXN_NEXT_TRANSACTION (40) XMXNC 446
XM_TXN_NONE 1 XMXNC 447
XM_TXN_NULL_ATTACH_MESSAGE 1 XMXNC 447
XM_TXN_NULL_DEFERRED_ABEND 448
XM_TXN_NULL_TOKEN 448
XM_TXN_ORIGINAL_TRANSACTION_ID (48) XMXNC 446
XM_TXN_PG_TOKEN (A8) XMXNC 446
XM_TXN_PRE_SCHEDULE 1 XMXNC 447
XM_TXN_PREV_TRANSACTION (44) XMXNC 446
XM_TXN_PRIMARY_CLIENT_REQUEST_BLOCK (1C) XMXNC 445
XM_TXN_PRIMARY_CLIENT_REQUEST_BLOCK_ADDR (1C) XMXNC 445
XM_TXN_PRIMARY_CLIENT_REQUEST_BLOCK_LEN (20) XMXNC 445
XM_TXN_PRIMARY_CLIENT_TYPE (13D) XMXNC 447
XM_TXN_PRIMARY_TRANSACTION_ID (70) XMXNC 446
XM_TXN_PRIORITY_SET (BIT) XMXNC 445
XM_TXN_PROHIBIT_INLINE_CALLS (BIT) XMXNC 445
XM_TXN_RE_ATTACHED_TRANSACTION (133) XMXNC 447
XM_TXN_RE_ATTACHED_UOW_TOKEN (7C) XMXNC 446
XM_TXN_REMOTE_NAME (2C) XMXNC 446
XM_TXN_REMOTE_SYSTEM (34) XMXNC 446
XM_TXN_REPORT_CONDITION (BIT) XMXNC 445
XM_TXN_RESTART (135) XMXNC 447

XM_TXN_RESTART_COUNT (7A) XMXNC 446
XM_TXN_RM_TOKEN (F0) XMXNC 447
XM_TXN_ROLLBACK_REQUESTED (134) XMXNC 447
XM_TXN_ROUTABLE_STATUS 447
XM_TXN_RRS_UR 1 XMXNC 447
XM_TXN_SCHEDULE_STAGE (68) XMXNC 446
XM_TXN_SCHEDULER 1 XMXNC 447
XM_TXN_SCHEDULER_ERROR_CHAIN (100) XMXNC 447
XM_TXN_SCHEDULER_RETRY_CHAIN (100) XMXNC 447
XM_TXN_SM_TOKEN (90) XMXNC 446
XM_TXN_SO_TOKEN (C0) XMXNC 447
XM_TXN_SOCKET 1 XMXNC 447
XM_TXN_START 1 XMXNC 447
XM_TXN_START_CODE (11) XMXNC 445
XM_TXN_START_TERMINAL 1 XMXNC 447
XM_TXN_SYSTEM_TRANSACTION (79) XMXNC 446
XM_TXN_TASK_PRIORITY (12) XMXNC 445
XM_TXN_TCLASS (BIT) XMXNC 445
XM_TXN_TCLASS_DELAY_ADDR 447
XM_TXN_TCLASS_LOCKED (BIT) XMXNC 445
XM_TXN_TCLASS_SCHEDULED 1 XMXNC 447
XM_TXN_TCLASS_TOKEN (10C) XMXNC 447
XM_TXN_TCLASS_WAIT_START (58) XMXNC 446
XM_TXN_TCLASS_WAIT_TIME (58) XMXNC 446
XM_TXN_TD_TOKEN (98) XMXNC 446
XM_TXN_TERM_PURGE_PROTECT (BIT) XMXNC 445
XM_TXN_TERMINAL 1 XMXNC 447
XM_TXN_TF_TOKEN (E8) XMXNC 447
XM_TXN_TOKEN 446
XM_TXN_TOKEN_OWNERS 448
XM_TXN_TRANDATA 1 XMXNC 447
XM_TXN_TRANDEF_TOKEN (80) XMXNC 446
XM_TXN_TRANNUM (3C) XMXNC 446
XM_TXN_TRANSACTION_ADDR (38) XMXNC 446
XM_TXN_TRANSACTION_GROUP_ID (13E) XMXNC 447
XM_TXN_TRANSACTION_TOKEN (38) XMXNC 446
XM_TXN_UOW_ID_SUPPLIED (BIT) XMXNC 445
XM_TXN_US_TOKEN (D8) XMXNC 447
XM_TXN_WB_TOKEN (C8) XMXNC 447
XM_TXN_WEB 1 XMXNC 447
XM_TXN_XM_RUN_TRANSACTION 1 XMXNC 447
XM_TXN_XM_TOKEN 447
XM_TXN_XS_TOKEN (D0) XMXNC 447
XM_XB (0) XMXBC 441
XM_XB_BROWSING_TXN 441
XM_XB_EYECATCHER (2) XMXBC 441
XM_XB_FLAGS (18) XMXBC 441
XM_XB_LENGTH 441
XM_XB_NEXT_XB (10) XMXBC 441
XM_XB_PREV_TXN (14) XMXBC 441
XM_XB_TOKEN_BROWSE (BIT) XMXBC 441
XM_XB_TOKEN_OWNER (19) XMXBC 441
XMA_ATTACH_COUNT (98) XMANC 436
XMA_CATALOG_LOCK_TOKEN (24) XMANC 435
XMA_CATALOGUED_STATE (C8) XMANC 436
XMA_CSXM_TRANDEF_TOKEN (9C) XMANC 436
XMA_CUSHION_SIZE_ABOVE (E4) XMANC 437
XMA_CUSHION_SIZE_BELOW (E0) XMANC 437
XMA_DETACH_COUNT (70) XMANC 436
XMA_DTRTRAN_TOKEN (64) XMANC 436
XMA_DTRTRAN_TOKEN_N (68) XMANC 436
XMA_DTRTRAN_TOKEN_P (64) XMANC 436
XMA_DTRTRAN_TRAN_ID (6C) XMANC 436
XMA_EYECATCHER (2) XMANC 435
XMA_FIRST_BAD_TXN_ENVIRONMENT (8C) XMANC 436
XMA_FIRST_TRANSACTION (74) XMANC 436
XMA_FIRST_TXN_BROWSE (7C) XMANC 436
XMA_FLAGS 435
XMA_FORCE_PURGE_ISSUED (BIT) XMANC 435
XMA_GENERAL_SUBPOOL (10) XMANC 435
XMA_GENERAL_SUBPOOL_24 (100) XMANC 437
XMA_GLOBAL_USER_EXITS_STATUS (20) XMANC 435
XMA_HIGH_TRANNUM (94) XMANC 436
XMA_LAST_RESET_TIME 437
XMA_LAST_TRANSACTION (78) XMANC 436
XMA_LENGTH 435
XMA_LOCAL_SYSTEM (40) XMANC 435
XMA_LOCK_TOKEN (18) XMANC 435
XMA_LOW_TRANNUM (90) XMANC 436
XMA_MXT_FLAGS (D4) XMANC 437
XMA_MXT_LIMIT (C8) XMANC 436
XMA_MXT_LIMIT_SET (BIT) XMANC 437
XMA_MXT_QUEUING (BIT) XMANC 437
XMA_MXT_TCLASS_PTR (CC) XMANC 437
XMA_MXT_TCLASS_TOKEN (CC) XMANC 436
XMA_PROFORMA_TXN (88) XMANC 436
XMA_RTXD_DIRECTORY_TOKEN (54) XMANC 435
XMA_RUNTRAN_SUBPOOL 437
XMA_SCHEDULER_ERROR_HEAD 437
XMA_STATIC_BLOCK_HEAD (44) XMANC 435
XMA_STATIC_BLOCK_TAIL (48) XMANC 435
XMA_STATS_BUFFER_PTR (F0) XMANC 437
XMA_SYSTEM_ATTACH_RETRY_HEAD (DC) XMANC 437
XMA_TCLASS_CHAIN_HEAD 436
XMA_TCLASS_CHAIN_TAIL (C0) XMANC 436
XMA_TCLASS_CONTROL_FLAGS (B8) XMANC 436
XMA_TCLASS_DIRECTORY_TOKEN (B0) XMANC 436
XMA_TCLASS_INSTANCE_COUNT (B4) XMANC 436
XMA_TCLASS_RECOVERY_COMPLETE (BIT) XMANC 436
XMA_TCLASS_SUBPOOL (A8) XMANC 436
XMA_TOTAL_TASKS (E8) XMANC 437
XMA_TPNM_DIRECTORY_TOKEN (58) XMANC 435
XMA_TRANDEF_CONTROL_FLAGS (4C) XMANC 435
XMA_TRANDEF_DIRECTORY_TOKENS (50) XMANC 435
XMA_TRANDEF_GLOBAL_STATE (28) XMANC 435
XMA_TRANDEF_INSTANCE_COUNT (60) XMANC 436
XMA_TRANDEF_INSTANCE_SUBPOOL (28) XMANC 435
XMA_TRANDEF_LOCK_TOKEN (5C) XMANC 435
XMA_TRANDEF_STATIC_SUBPOOL (30) XMANC 435
XMA_TRANDEF_SUBPOOL_TOKENS (28) XMANC 435
XMA_TRANDEF_TPNAME_SUBPOOL (38) XMANC 435
XMA_TRANNUM_RANGE (90) XMANC 436
XMA_TRANSACTION_GLOBAL_STATE (70) XMANC 436
XMA_TRANSACTION_SUBPOOL (80) XMANC 436
XMA_TXD_DIRECTORY_TOKEN (50) XMANC 435
XMA_TXD_RECOVERY_COMPLETE (BIT) XMANC 435
XMA_TXN_WAITING_FOREVER (BIT) XMANC 435
XMA_XM_STATE (1C) XMANC 435
XMA_XRSINDI_ACTIVE (BIT) XMANC 435
XMA_XXMATT_ACTIVE (BIT) XMANC 435
XMANC 435
XMANCHOR (0) XMANC 435
XMCAT 438
XMCLC 439
XMEOUT_ACTIVE (BIT) MEPS 257
XMRLC 440
XMXBC 441
XMXDC 441
XMXNC 445
xpi
  inquire application data xpi command, APIQ 2
XRH (0) TSAUX 388
XRH_DATA (24) TSAUX 389
XRH_FLAGS (20) TSAUX 388
XRH_FMH (BIT) TSAUX 389
XRH_ITEM_NUMBER (4) TSAUX 388
XRH_LENGTH (0) TSAUX 388
XRH_QUEUE_NAME (10) TSAUX 388
XRH_RECOVERABLE (BIT) TSAUX 389
XRH_REQUIRED (BIT) TSAUX 389
XRH_SECTION_LENGTH (22) TSAUX 389
XRH_SECTION_NUMBER (6) TSAUX 388
XRH_TIME_STAMP (8) TSAUX 388
XS_DOMAIN_LOCKNAME 8 XSANC 451
XS_EXTRACT_LOCKNAME 8 XSANC 451
XS_REBUILD_LOCKNAME 8 XSANC 451
XS_RESCHECK_LOCKNAME 8 XSANC 451
XS_STATE_INITIALISED 1 XSANC 449
XS_STATE_INITIALISING 1 XSANC 449
XS_STATE_QUIESCED 1 XSANC 449
XS_STATE_QUIESCING 1 XSANC 449
XS_STATE_TERMINATED 1 XSANC 449
XSA (0) XSANC 448
XSA_APPC_SEED (18) XSANC 448
XSA_AUTHORIZED_BLOCK_POINTER (14) XSANC 448
XSA_CICS_SVC 448
XSA_CICS_SVC_NUMBER (13) XSANC 448
XSA_CICS_SVC_OPCODE (12) XSANC 448
XSA_DOMAIN_LOCK_TOKEN (2C) XSANC 448
XSA_EXTRACT_LOCK_TOKEN (38) XSANC 448
XSA_EYE_CATCHER 14 XSANC 451
XSA_PREFIX (0) XSANC 448
XSA_PREFIX_LENGTH (0) XSANC 448
XSA_PREFIX_TEXT (2) XSANC 448
XSA_REBUILD_LOCK_TOKEN (34) XSANC 448
XSA_RESCHECK_LOCK_TOKEN (30) XSANC 448

XSA_SPTOKEN_GENERAL (1C) XSANC 448
XSA_XS_STATE (10) XSANC 448
XSA_XSXM_POOL (24) XSANC 448
XSANC 448
XSDI_ACEE_PTR (18) XSSS 455
XSDI_APPLID 455
XSDI_APPLID_X 454
XSDI_ENTRY_PORT 455
XSDI_FLAGS (2) XSSS 454
XSDI_LENGTH (0) XSSS 454
XSDI_SECURITY_ENTRY (0) XSSS 454
XSDI_USERID 454
XSSS 451
XSSS_APPC (A8) XSSS 452
XSSS_APPCLU_FILTER (40) XSSS 452
XSSS_APPCLU_FILTER_LENGTH (40) XSSS 452
XSSS_APPCLU_FILTER_STRING (42) XSSS 452
XSSS_ARROW (2) XSSS 451
XSSS_BLOCKID (8) XSSS 451
XSSS_CLASSNAME_COUNT 452
XSSS_CLASSNAME_TABLE (A8) XSSS 452
XSSS_CLASSNAME_TABLE_END (120) XSSS 454
XSSS_CMDSEC (BIT) XSSS 451
XSSS_COMPONENT (3) XSSS 451
XSSS_CWA_ADDRESS 451
XSSS_DB2ENTRY (C6) XSSS 453
XSSS_DEFAULT_SECURITY_TOKEN 451
XSSS_DIRECTORY_PTR (94) XSSS 452
XSSS_EARLY_VERIFY_ROUTINE (20) XSSS 451
XSSS_EXTENSION_MANAGER_PTR (A0) XSSS 452
XSSS_EYECATCHER (0) XSSS 451
XSSS_FILE (DA) XSSS 453
XSSS_FLAG1 (11) XSSS 451
XSSS_FLAG2 451
XSSS_FLAG3 451
XSSS_FLATTENED_SECURITY_LENGTH 1 XSSS 455
XSSS_GENERIC_APPLID (58) XSSS 452
XSSS_INSTLN_REQUIRED (BIT) XSSS 451
XSSS_JOBSTEP_SECURITY_TOKEN (38) XSSS 451
XSSS_JOURNAL (E4) XSSS 453
XSSS_LENGTH (0) XSSS 451
XSSS_PARTNER_CHECK (BIT) XSSS 451
XSSS_PREFIX 452
XSSS_PREFIX_REQUIRED (BIT) XSSS 451
XSSS_PROGRAM (EE) XSSS 453
XSSS_PSB (F8) XSSS 453
XSSS_PSB_CHECK (BIT) XSSS 451
XSSS_REGION_GROUPID 452
XSSS_REGION_USERID 452
XSSS_RESSEC (BIT) XSSS 451
XSSS_SECURITY_ACTIVE (BIT) XSSS 451
XSSS_SECURITY_TOKEN_MANAGER 452
XSSS_SECURITY_VECTOR_TABLE (20) XSSS 451
XSSS_SPCOMMAND (BC) XSSS 453
XSSS_STORAGE_INTERFACE_PTR (98) XSSS 452
XSSS_STORAGE_MANAGER_PTR (9C) XSSS 452
XSSS_SUBSYS (18) XSSS 451
XSSS_SURROGATE (116) XSSS 454
XSSS_SURROGATE_CHECK (BIT) XSSS 451
XSSS_TDQUEUE (D0) XSSS 453
XSSS_TOKEN_HWMK (A4) XSSS 452
XSSS_TRANSACTION (B2) XSSS 452
XSSS_TRANSATTACH (10C) XSSS 454
XSSS_TSQUEUE (102) XSSS 453
XSSS_V321 1 XSSS 455
XSSS_V410 1 XSSS 455
XSSS_VERSION (10) XSSS 451
XSSS_VERSION_NUM 1 XSSS 455
XSXD 455
XSXD_COMMUNICATION_AREA (30) XSXD 455
XSXD_EDF_TOKEN (10) XSXD 455
XSXD_FACILITY_TOKEN 455
XSXD_PRINCIPAL_TOKEN 455
XSXD_SESSION_TOKEN (8) XSXD 455
XSXD_TRANSACTION_DATA (0) XSXD 455
XSXD_UNIQUE_TOKEN (18) XSXD 455
XSXD_UNIQUE_TOKEN_LIST (18) XSXD 455
XSXM_SUBPOOL_NAME 8 XSANC 450
XSXT 456
XSXT_CMDSEC (BIT) XSXT 456
XSXT_COUNT (6) XSXT 456
XSXT_RESSEC (BIT) XSXT 456
XSXT_STACK (4) XSXT 456
XSXT_STACK_1 (4) XSXT 456
XSXT_STACK_2 (5) XSXT 456
XSXT_TRAN_DATA_PTR (0) XSXT 456
XSXT_TRAN_TOKEN (0) XSXT 456

Y

YES 0 MEPS 259
YES 0 PAA 284
YES 0 TIA 380

Z

Z_ANCHOR (98) DSANC 55
Z_NUMBER (9C) DSANC 55
ZBMEXVAL 4 TSAUX 389
ZCQ 456
ZEMPTY 4 TSAUX 389
ZMINREF 4 TSAUX 389
ZSUPP_NO 1 MEPS 259
ZSUPP_YES 1 MEPS 259

Sending your comments to IBM

CICS® Transaction Server for OS/390®

CICS Supplementary Data Areas

LY33-6090-02

If you want to send to IBM any comments you have about this book, please use one of the methods listed below. Feel free to comment on anything you regard as a specific error or omission in the subject matter, and on the clarity, organization or completeness of the book itself.

To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate, without incurring any obligation to you.

You can send your comments to IBM in any of the following ways:

- By mail:

  IBM UK Laboratories
  Information Development
  Mail Point 095
  Hursley Park
  Winchester, SO21 2JN
  England

- By fax:
  - From outside the U.K., after your international access code use 44 1962 870229
  - From within the U.K., use 01962 870229
- Electronically, use the appropriate network ID:
  - IBM Mail Exchange: GBIBM2Q9 at IBMMAIL
  - IBMLink: HURSLEY(IDRCF)
  - Email: idrcf@hursley.ibm.com

Whichever method you use, ensure that you include:

- The publication number and title
- The page number or topic to which your comment applies
- Your name and address/telephone number/fax number/network ID.

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

Spine information:

IBM CICS TS for OS/390 CICS Supplementary Data Areas *Release 3*