E06

Exploring Database Recovery Control

Pat Schroeck

IMS

technical conference

Las Vegas, NV **September 15 - September 18, 2003**

RETURN TO INDEX

Agenda

- DBRC - what and why
- DBRC components
- DBRC tasks
- The RECON
- DBRC commands
- IMS calls to DBRC
- DBRC diagnostics
- V8 enhancement overview

What is DBRC?

- IMS **D**ata **B**ase **R**ecovery **C**ontrol facility
  - Logging and database recovery.
  - Database access in a data sharing environment.
  - Generates JCL for various IMS Utilities
    - Database Recovery
    - Change Accumulation
    - Image Copy
    - Log Archive
    - Log Recovery

DBRC, the early days

- The IMS Backup and Recovery Feature
  - Supported the main IMS Utilities
    - Data Base Recovery
    - Change Accumulation
    - Image Copy
- DBRC Release 1 shipped December 1978 with IMS 1.1.4
  - Prior to IMS 6.1, DBRC would reject commands with times < 1978

Why DBRC?

- Originally, DBRC was an optional, separately priced feature of IMS

- It controls the data sharing environment
  - introduced in 1981 by IMS 1.2 (DBRC Release 2)

- DBRC plays a key role in managing the log data needed to restart and recover IMS online subsystems
  - DASD logging introduced in 1984 by IMS 1.3 (DBRC Release 3)

When to Use DBRC

- These IMS environments require DBRC
  - Online: DB / DC, DCCTL, or DBCTL
  - Data sharing environments, including IMS Sysplex
  - Extended Recovery Facility (XRF) - IMS 2.1
  - Remote Site Recovery (RSR) - IMS 5.1
  - HALDB (IMS 7.1)

- When you want to simplify your recovery process and protect your databases

DBRC Components

- The **Re**covery **Con**trol (**RECON**) datasets
- Command support through the Database Recovery Control utility (DSPURX00)
- Skeletal Job Control Language (JCL) for various IMS recovery utilities

DBRC Tasks

- Automatic tasks performed thru DBRC and IMS interaction
  - Log control for IMS
  - Recording recovery information in the RECON
  - Verifying that IMS database utilities have correct JCL input
  - Controlling the recovery of registered DBs
  - Controlling the IMS data sharing environment

- Command requested tasks
  - Recording recovery information in the RECON
  - JCL generation for IMS utilities
  - Listing information in the RECON

The RECON Data Sets

- DBRC's primary resource
  - Keep status for the IMS System and DB activity
    - for Backup and Recovery
  - Control access and activity
    - IMS logs
    - IMS databases
    - Data sharing
- VSAM KSDSs
  - 32-byte Key
- Define them using the Access Method Services DEFINE CLUSTER statement
- Use 3 RECONs to maintain dual-mode operation

Creating the RECON with Access Method Services

```
//*      DELETE AND DEFINE RECON DATA SET
//DEFRCN  EXEC PGM=IDCAMS
//DD1       DD VOL=SER=IAPP01,DISP=SHR,UNIT=3390
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD *
 DELETE (IMS61.RECON1 IMS61.RECON2 IMS61.RECON3 )
 SET MAXCC = 0
 DEFINE CLUSTER (NAME(IMS61.RECON1) FILE(DD1) -
     INDEXED KEYS(32 0) VOL(IAPP01) TRK(3 3) FSPC(20 20) -
     SHR(3 3) SPANNED NOREUSE UNORDERED )
 DEFINE CLUSTER (NAME(IMS61.RECON2) FILE(DD1) -
     INDEXED KEYS(32 0) VOL(IAPP01) TRK(3 3) FSPC(20 20) -
     SHR(3 3) SPANNED NOREUSE UNORDERED )
  DEFINE CLUSTER (NAME(IMS61.RECON2) FILE(DD1) -
     INDEXED KEYS(32 0) VOL(IAPP01) TRK(3 3) FSPC(20 20) -
     SHR(3 3) SPANNED NOREUSE UNORDERED )
/*
```

RECON Initialization

- Initialize the RECON datasets after defining them with AMS
- Issue the DBRC INIT.RECON command
  - DBRC Utility - DSPURX00
  - RECON must be empty
  - Builds the RECON Header and Header Extension records
    - Control system options
    - Controls RECON management
      - COPY 1, COPY 2, SPARE

RECON Records

- RECON (Header, Header Extension)
- PRILOG, SECLOG, IPRI, ISEC
- PRIOLD, SECOLD, PRISLD, SECSLD, IPRIOL,ISECOL, IPRISL, ISECSL
- LOGALL
- GSG
- SSYS
- BACKOUT
- CAGRP
- CA
- DBDSGRP (DBGRP and RECOVGRP)
- DB
- DBDS
- ALLOC
- IMAGE
- REORG
- RECOV

DBRC Command Verbs

- **BACKUP**
  - Make a copy of the RECON Data Set
- **CHANGE**
  - Modify the contents an existing RECON record
- **DELETE**
  - Delete an existing RECON record
- **GENJCL**
  - Generate JCL for a utility or function
- **INIT**
  - Initialize (create) a RECON record
- **LIST**
  - Produce formatted output of selected RECON records
- **NOTIFY**
  - Add information to the RECON that is normally written automatically
- **RESET**
  - RSR-only

DBRC Command Syntax

- Standard, command language syntax
  - Similar to TSO and Access Method Services
- Consists of
  - Verb
  - Modifier
    - Keyword parameters - required/optional, defaults
      BACKUP.RECON
      LIST.RECON STATUS
- DBRC Time Stamps
  - Standard Compressed format
    - yydddhhmmsst [offset]

      LIST.LOG STARTIME(970231212121)
  - Standard Punctuated format
    - [yy]yy|ddd|hh|mm|ss|t

      LIST.LOG STARTIME('97.023 12:12:12.1 PST')

The LIST Command

- Use these LIST command modifiers to display existing RECON records.

  - BKOUT - Backout
  - CAGRP - CA Group
  - DB - Database
  - DBDS - Database dataset or data entry database
  - DBDSGRP - DBDS Group
  - GSG - Global Service Group
  - HISTORY - activity for DBDSs or areas
  - LOG - RLDS/SLDS, OLDS
  - SUBSYS - subsystem
  - RECON - full contents of the RECON or its STATUS

LIST.RECON Command

- LIST.RECON
  - Lists the contents of the entire RECON
  - Generally creates lots of output
  - DBRC holds the RECONs for the duration of the command - may take a long time

- LIST.RECON STATUS
  - the RECON record may contain all you need
  - generates less than a page of output
  - quick
  - the IMS MTO can issue via the /RM command

    /RML DBRC='RECON STATUS'

The INIT Command

- Use the INIT command to create a RECON record.
  - ADS
  - CA
  - CAGRP
  - DB
  - DBDS
  - DBDSGRP
  - GSG
    - SG
  - IC
  - PART
  - RECON

INIT.RECON Command

- The first command to issue after defining the RECON

- The RECON must be empty

- No required parameters, but defaults may suffice
  - CATDS | NOCATDS
  - FORCER | NOFORCER
  - LISTDL|NOLISTDL
  - ...

- Example:

```
INIT.RECON SSID(IMS3)
```

RECON Record Contents

- The RECON record is comprised of 2 records
  - RECON Header record
  - RECON Header Extension record
    - Internal processing information
- RECON Initialization Token
  - Used for DBRC Usage Indicator (DUI) processing
- Processing Options (e.g. FORCER)
  - Time Stamp Format Information
- Status indicators include:
  - Minimum Version (new in V8)
  - RECON DD Names
  - RECON Data Set Names
  - COPY1, COPY2, and SPARE

RECON Record Example (V8)

```
 RECON
  RECOVERY CONTROL DATA SET, IMS V8R1
  DMB#=11                            INIT TOKEN=02226F1953352F
  NOFORCER  LOG DSN CHECK=CHECK17    STARTNEW=NO
  TAPE UNIT=3400      DASD UNIT=SYSDA     TRACEOFF   SSID=**NULL**
  LIST DLOG=YES                CA/IC/LOG DATA SETS CATALOGED=NO
  MINIMUM VERSION = 6.1
  LOG RETENTION PERIOD=00.000 00:15:00.0
  COMMAND AUTH=NONE  HLQ=**NULL**
  SIZALERT DSNUM=15      VOLNUM=16     PERCENT= 95
  LOGALERT DSNUM=3       VOLNUM=16

  TIME STAMP INFORMATION:

    TIMEZIN = %SYS                -LABEL- -OFFSET-
                                   PDT      -07:00
                                   PST      -08:00

    OUTPUT FORMAT:  DEFAULT = LOCORG NONE   PUNC YY
 IMSPLEX = ** NONE **

-DDNAME-      -STATUS-      -DATA SET NAME-
 RECON1        COPY1         IMSTESTL.IMS.RECON1
 RECON2        COPY2         IMSTESTL.IMS.RECON2
 RECON3        SPARE         IMSTESTL.IMS.RECON3
```

RECON Record Example (V6)

```
RECON
 RECOVERY CONTROL DATA SET, IMS/ESA V6R1        COEXISTENCE ENABLED
 DMB#=8                             INIT TOKEN=A4060F2231139F
 NOFORCER  LOG DSN CHECK=CHECK17    STARTNEW=NO
 TAPE UNIT=3480      DASD UNIT=SYSDA     TRACEOFF   SSID=**NULL**
 LIST DLOG=NO                 CA/IC/LOG DATA SETS CATALOGED=NO
 LOG RETENTION PERIOD=00.001 00:00:00.0

 TIME STAMP INFORMATION:

   TIMEZIN = %SYS

   OUTPUT FORMAT:  DEFAULT = LOCORG NONE   PUNC YY
                   CURRENT = LOCORG NONE   PUNC YY

 -DDNAME-      -STATUS-       -DATA SET NAME-
  RECON1        COPY1          IMSTESTL.IMS.RECON1
  RECON2        COPY2          IMSTESTL.IMS.RECON2
  RECON3        UNAVAILABLE
THT
   -LOCAL START-   -OFFSET-
 0000.000 00:00:00.0 -08:00
```

INIT.DB Command

- Use the INIT.DB command to register a database with DBRC and to set the database sharing level desired (0,1,2,or 3)

- To register a High Availability Large Database (HALDB) use
  - ► The HALDB Partition Definition utility

    or

  - ► INIT.DB in conjunction with INIT.PART
    - – INIT.PART automatically creates the DBDS records

DB Record Contents

- Database Definition (DBD) name
- Status flags and counters
  - Backout Needed, Prohibit Authorization, IC needed, etc.
- Sharing status
  - Share Level
  - Global DMB number
  - IRLM ID
  - Authorized subsystems
- Recoverable status
  - Non-recoverable DEDB support added in IMS V8.
- RSR Information

DB Record Example

```
DB
 DBD=DHVNTZ02                                     DMB#=2       TYPE=IMS
 SHARE LEVEL=1               GSGNAME=**NULL**     USID=0000000001
 AUTHORIZED USID=0000000000  RECEIVE USID=0000000000 HARD USID=0000000000
 RECEIVE NEEDED USID=0000000000
 DBRCVGRP=**NULL**
 FLAGS:                             COUNTERS:
   BACKOUT NEEDED        =OFF         RECOVERY NEEDED COUNT   =0
   READ ONLY             =OFF         IMAGE COPY NEEDED COUNT =0
   PROHIBIT AUTHORIZATION=OFF         AUTHORIZED SUBSYSTEMS   =0
   RECOVERABLE           =YES         HELD AUTHORIZATION STATE=0
                                      EEQE COUNT              =0
   TRACKING SUSPENDED    =NO          RECEIVE REQUIRED COUNT  =0
   OFR REQUIRED          =NO
```

INIT.DBDS Command

- Use the INIT.DBDS command to register a DBDS or DEDB area
  - DBRC examines the IMS DBDLIB data set
    - to verify the DBDS or DEDB area exists
    - to obtain the DBDS's data set identifier (DSID), its database organization (DBORG), and its dataset organization (DSORG)
- Use its parameters to specify
  - names of your skeletal JCL members
  - IC GENMAX value
  - IC REUSE/NOREUSE
  - Recovery Period (RECOVPD)
- The INIT.DBDS command fails if you issue it while the DB is in use
- Not allowed for HALDBs

DBDS Record Contents

- DBD and DD names
- Data Set Name (DSN)
- Database type (e.g. IMS , HALDB, FP)
- DB and DS organization (e.g. DBORG=HDAM, DSORG=VSAM)
- Status flags and counters
  - IC NEEDED, RECOV NEEDED
  - Extented Error Queue Element (EEQE) count
- Image Copy information
  - REUSE or NOREUSE, GENMAX, IC AVAIL, IC USED
- Recovery information
  - RECOV NEEDED, RECOVPD
- Change Accumulation Group name
- Skeletal JCL execution member names
  - ICJCL, OICJCL, RECOVJCL, etc.

DBDS Record Example

```
DBDS
 DSN=DHVNTZ02.FKXXI02E                                          TYPE=IMS
 DBD=DHVNTZ02  DDN=HIDAM2   DSID=002 DBORG=HIDAM  DSORG=VSAM
 CAGRP=**NULL**  GENMAX=2     IC AVAIL=0     IC USED=0     DSSN=00000000
 NOREUSE         RECOVPD=0       OTHER DDN=**NULL**
 DEFLTJCL=**NULL**  ICJCL=ICJCL     OICJCL=OICJCL    RECOVJCL=RECOVJCL
 RECVJCL=ICRCVJCL
 FLAGS:                             COUNTERS:
   IC NEEDED      =OFF
   IC RECOMMENDED =ON
   RECOV NEEDED   =OFF
   RECEIVE NEEDED =OFF                EEQE COUNT               =0
```

HALDB Record Contents

- Master name
- Share level
- Recoverable status
- DB and DS organization
- Partition Selection Routine (optional)
- Change version number
- Current Partition ID
- Partition count
- Number of DSGs
- GSG name (RSR)

HALDB Record Example

```
DB
 DBD=DBHDOJ01                        DMB#=2      CHANGE#=5     TYPE=HALDB
 SHARE LEVEL=0               GSGNAME=**NULL**
 PSNAME=**NULL**  DBORG=PHDAM    DSORG=OSAM    CURRENT PARTITION ID=00004
 FLAGS:                             COUNTERS:
   RECOVERABLE           =YES         PARTITIONS              =4
                                      DATA SET GROUP MEMBERS  =1
```

INIT.PART Command

- Use INIT.PART command to register a HALDB Partition
  - DBRC creates the RECON HALDB partition structure
    - a PART record
    - the partition DB record
    - one or more DBDS records (determined by the DBD specification)
      - similar to a non-HALDB DBDS
    - TYPE=PART
  - Specify
    - names of skeletal JCL members
    - IC REUSE or NOREUSE
    - IC GENMAX value
- Command fails if the HALDB is being used by the HALDB Partition Definition utility

DB PART Record Contents

- Similar to a non-HALDB DB record, but with more partition related fields
- Partition High Key / String in character and hexidecimal forms
- Randomizer information
- Free Space information
- OSAM Block Size
- Current sharing status (i.e. authorized subsystems)

DB PART Record Example

```
DB
 DBD=PDHDOJA   MASTER DB=DBHDOJ01                CHANGE#=2     TYPE=PART
 USID=0000000002  AUTHORIZED USID=0000000002  HARD USID=0000000002
 RECEIVE USID=0000000002  RECEIVE NEEDED USID=0000000000
 DBRCVGRP=**NULL**
 DSN PREFIX=IMSTESTS.DBHDOJ01                          PARTITION ID=00001
 PREVIOUS PARTITION=**NULL**    NEXT PARTITION=PDHDOJB
 OLRIMSID=**NULL**    ACTIVE DBDS=A-J      M-V EXIST=NO
 RANDOMIZER:
   NAME=DFSHDC20  ANCHOR=3    HIGH BLOCK#=3         BYTES=25
 FREE SPACE:
   FREE BLOCK FREQ FACTOR=0    FREE SPACE PERCENTAGE=0
 PARTITION HIGH KEY/STRING (CHAR):               (LENGTH=5  )
   J0200
 PARTITION HIGH KEY/STRING (HEX):
   D1F0F2F040404040404040404040404040404040404040404040404040404040

 OSAM BLOCK SIZE:
   A = 4096

 FLAGS:                             COUNTERS:
   BACKOUT NEEDED        =OFF         RECOVERY NEEDED COUNT   =0
   READ ONLY             =OFF         IMAGE COPY NEEDED COUNT =0
   PROHIBIT AUTHORIZATION=OFF         AUTHORIZED SUBSYSTEMS   =0
                                      HELD AUTHORIZATION STATE=0
                                      EEQE COUNT              =0
   TRACKING SUSPENDED    =NO          RECEIVE REQUIRED COUNT  =0
   OFR REQUIRED          =NO
   PARTITION INIT NEEDED =NO
   ONLINE REORG ACTIVE   =NO
   PARTITION DISABLED    =NO
```

INIT.DBDSGRP Command

- Use an INIT.DBDSGRP command to define a group
  - **DB Group**
    - DL/I databases or DEDB areas
    - use the DBGRP parameter
  - **DBDS Group**
    - DBDSs or DEDB areas
    - use the MEMBERS parameter
  - **Recovery Group**
    - DL/I databases or DEDB areas
    - used with Database Recovery Facility (DRF) recoveries
    - use the RECOVGRP parameter
    - A DB/Area can only belong to one Recovery Group

Group Record Examples

```
DBGRP
 GRPNAME=DBGRP1               #MEMBERS=6    -DBD/AREA-
                                             DIVNTZ02
                                             DHVNTZ02
                                             DXVNTZ02
                                             DB21AR0
                                             DB21AR1
                                             DB21AR2

DBDSGRP
 GRPNAME=FJKGRP               #MEMBERS=5     -DBD-    -DDN/AREA-
                                             DIVNTZ02  DBHVSAM1
                                             DIVNTZ02  DBHVSAM2
                                             DHVNTZ02  HIDAM
                                             DHVNTZ02  HIDAM2
                                             DXVNTZ02  XDLBT04I

RECOVGRP
 GRPNAME=RCVGRP1              #MEMBERS=5     -DBD-     -AREA-
                                             DIVNTZ02
                                             DHVNTZ02
                                             DXVNTZ02
                                             DEDBJN21  DB21AR0
                                             DEDBJN21  DB21AR1
```

INIT.CAGRP Command

- INIT.CAGRP command to specify the DBDSs you want to belong to a specific Change Accumulation Group

- You must have done an INIT.DBDS for each of the DBDSs you identify on the INIT.CAGRP command

- A DBDS can only belong to one CA Group

CAGRP Record Contents

- CA group name
- Status flags, counters
- Name of the CA skeletal JCL for GENJCL
- Names of the DBDS members of the group

CAGRP Record Example

```
CAGRP
 GRPNAME=CAGRP1   GRPMAX=3    CA AVAIL=0    CA USED=1
 NOREUSE CAJCL=CAJCL          DEFLTJCL=**NULL**
                              #MEMBERS=4    -DBD-     -DDN-
                                            DEDBJN21  DB21AR1
                                            DEDBJN21  DB21AR3
                                            DEDBJN21  DB21AR6
                                            DEDBJN21  DB21AR7
```

INIT.IC Command

- Use the INIT.IC command to create an IMAGE copy record in the RECON to define the image copy dateset available for use during a subsequent execution of a supported Image Copy Utility.
  - DBDS must have the REUSE attribute
- Each INIT.IC command creates one IMAGE copy record
  - You can specify a duplicate image copy dataset

Available IMAGE Record Example

```
                             AVAILABLE DATA SET
                          DBD=DEDBJN21 DDN=DB21AR11

 IMAGE
* CREATE  = 03.190   09:13:09.7*

 IC1
  DSN=IMSVS.DEDBJN21.DB21AR11.IC.IC091307          FILE SEQ=0001
  UNIT=3400                          VOLS DEF=0000 VOLS USED=0000
```

INIT.GSG Command

- Use the INIT.GSG command to define a Global Service Group (GSG) in a RECON
- Used for Remote Site Recovery (RSR) functions
- Command fails if RSRFEAT=NO is specified in IMSCTRL macro
- Must define the GSG to every RECON used by an IMS subsystem in the Global Service Group

INIT.SG Command

- INIT.SG command to define a service group as a member of a Global Service Group (GSG)

- Used for Remote Site Recovery (RSR) functions

GSG Record Contents

- GSG name
- Service group names and roles
- Time stamps
  - Start time of current log
  - Highest active site time
- Status flags and counters

GSG Record Example

```
GSG
 GSGNAME=IMSGSG1      #SGS=2      -SGNAME-      -ROLE-
                                  STLSITE1     ACTIVE       LOCAL
                                  STLSITE2     TRACKING
 CURRENT PRILOG TOKEN      = 6           TAKEOVER TOKEN = 0
 MINIMUM PRILOG TOKEN      = 1           DSN SEQ NUMBER = 0
 START TIME OF CURRENT LOG = 02.226 11:49:19.2
 HIGHEST ACTIVE SITE TIME  = 00.000 00:00:00.0
 TRACKING SUBSYSTEM ID     = **NULL**
 TAKEOVER IN PROGRESS
```

The CHANGE Commands

- Use the CHANGE command with these Modifiers to alter an existing RECON record.
  - **ADS** - Area Dataset
  - **BKOUT**
  - **RECON**
    - CATDS, TRACEON or TRACEOFF,LOGALERT, SIZALERT
    - LISTDL|NOLISTDL
    - UPGRADE (introduced in V7)
  - **DB**
  - **DBDS**
  - **DBDSGRP**
  - **CA** - Change Accumulation
    - INVALID
  - **CAGRP**
  - **IC or UIC** - Image Copy, User IC
  - **PART** - V7/V8 SPEs
  - **PRILOG (SECLOG)**
    - ERROR
  - **SUBSYS**

GENJCL Command

- Use the GENJCL command to generate the JCL and utility control statements to execute these IMS utilities
  - GENJCL.ARCHIVE
    - Log Archive
  - GENJCL.CA
    - CA utility
  - GENJCL.CLOSE
    - Log Recovery utility
  - GENJCL.IC or OIC
    - Image Copy or Online Image Copy utility
  - GENJCL.RECEIVE
    - Database Recovery utility
  - GENJCL.RECOV
    - Database Recovery utility

- Use the GENJCL.USER command to generate JCL or any kind of user determined output

Skeletal JCL

- DBRC provides the following Skeletal JCL execution Partitioned Data Set (PDS) members and uses them as a model for performing keyword substitution and JCL generation.

  - ARCHJCL - for the IMS Log Archive Utility (DFSUARC0)
  - CAJCL - for the IMS Database Change Accumulation utility (DFSUCUM0)
  - LOGCLJCL - for the IMS Log Recovery utility (DFSULTR0) to close a log
  - ICJCL - for the IMS Database Image Copy utility (DFSUDMP0)
  - OICJCL - for the IMS Online Image Copy utility (DFSUICP0)
  - ICRCVJCL - for GENJCL.RECEIVE to generate JCL to run the IMS Database Recovery utility for image copy receive
  - RECOVJCL - for the IMS Database Recovery utility (DFSURDB0).
  - JOBJCL - single JCL statement you modify with your installation's requirements
  - DSPUPJCL also provided
    - User JCL to rebuild the Index and/or ILE data sets for a HALDB Partition
  - User Options:
    - modify supplied skeletal member slightly
    - create own skeletal member

The BACKUP Command

- Use the BACKUP command to create a backup copy of the RECON.

```
BACKUP.RECON RECON1 | RECON2 | BOTH
```

- The command invokes IDCAMS REPRO
  - **Note**: Backup to a sequential data set has a 32K record size restriction

IMS calls to DBRC

- IMS and DBRC need to communicate in a number of environments
  - Batch
  - Online
  - Utilities
- Communication is done via macros
  - DFSBRLSC, the IMS-DBRC Interface
  - DFSBRLSB, the common parameter block
- Typical Functions
  - DBRC Initialization, Termination
  - Signon, Signoff
  - Log Processing (Open, Close,EOV)
  - Archive
  - DB Authorization, Open, Allocation
  - Backout

DFSBRLSC Functions and DBRC Processing

- DBRC Processing depends on the DFSBRLSC FUNCTION
  - DFSBRLSC ID=DBA
    - FUNC= SIGNON or SIGNOFF
      - Subsystem signon/off processing (DSPSSIGN)
    - FUNC=ALLOC or DEALLOC
      - DBDS Allocation/Deallocation processing (DSPALD00)
  - DFSBRLSC ID=DLG
    - FUNC=OPEN, Switch, or Close
      - DASD Log processing (DSPOLD00)
  - DFSBRLSC ID=ARC
    - FUNC=INIT, EOJ, ARCOMPL
    - Archive processing (DSPARC00)

DFSBRLSC Functions and DBRC Processing ...

- DFSBRLSC ID=ICU
  - FUNC= INIT, EOD, EOJ
  - Image copy utility processing
- DFSBRLSC ID=CAU
  - FUNC=INIT, EOJ
  - Change accumulation utility (DFSUCUM0) processing
- DFSBRLSC ID=RVU
  - FUNC= INIT, EOJ
  - Database recovery utility (DFSURDB0) processing
- DFSBRLSC ID=DBA
  - FUNC=ALLOC
  - Allocation processing

DFSBRLSC Functions and DBRC Processing ...

- DFSBRLSC ID= REN
  - FUNC= INIT, EOJ
    - DB Unload utility processing
  - FUNC= INIT, EOJ/ INIT, REORG, EOJ
    - HD/HISAM Reorg/Reload utility processing
- DFSBRLSC ID = ; FUNC= UNAUTH; ALTID = REN
  - DB Scan utility (DFSURGS0) processing
- DFSBRLSC ID=BOU
  - FUNC= INIT, LOGSET, SAVUOR, VERUOR, UPDUOR, EOJ
    - Batch Backout utility (DFSBBO00) processing
- DFSBRLSC ID= SLR
  - FUNC= INIT, OPEN, CLOSE, EOJ
    - Log Recovery utility (DFSULTR0) processing

DBRC Exit Processing (Initial)

The IMS-DBRC Interface Macro, DFSBRLSC, provides a number of functions that drive specific DBRC processing. In general, DBRC Exit Processing follows this flow on the "Initial" call:

- ALLOCATE the RECONs
- RESERVE the RECONs
  - Done in DD Name sequence
  - QNAME is DSPURI01; RNAME is RECON DSN
- OPEN the RECONs
- VERIFY if needed
- READ RECON Header and Header Extension for each RECON
  - Determine Copy 1, 2, and Spare
  - Reconfigure if needed and possible
  - DEALLOCATE unused RECON datasets
- Cleanup the RECON if needed
- Perform requested DFSBRLSC Function
- Release the active RECONs

DBRC Exit Processing (Subsequent)

In general, DBRC Exit Processing follows this flow on subsequent calls:

- RESERVE the RECONs
  - Done in DD Name sequence
  - QNAME is DSPURI01; RNAME is RECON DSN
- Invalidate buffers
- READ RECON Header and Header Extension for each RECON
  - Determine Copy 1, 2, and Spare
  - Reconfigure if needed and possible
  - DEALLOCATE unused RECON datasets
- VERIFY if needed
- Cleanup the RECON if needed
- Perform requested DFSBRLSC Function
- If Function=TERM then CLOSE RECONs
- Release the active RECONs

SIGNON and SIGNOFF Processing

- SIGNON
  - Normal
  - Signon Recovery

- SIGNOFF
  - Normal
    - DBs unauthorized
    - DBRC deletes the SSYS record
  - Abnormal
    - DBRC sets the ABNORMAL TERM flag in the SSYS record
    - DB authorizations held

Normal Signon

- Build SSYS record for the subsystem
  - Fails if SSYS record already exists in the RECON
- Driven by:
  - DLI/DBB batch initialization
    - Subsystem name is the job name
  - /NRE online processing
    - Subsystem name is the IMSID name

Signon Recovery

- Signon Recovery Start
  - Fails if SSYS record already exists in the RECON (unless IMS restart)
  - Sets Recovery Started bit in SSYS record for the subsystem
- Signon Recovery End
  - Releases data base authorizations
  - When complete, environment looks like Signon Normal
- Driven by:
  - /ERE online processing
    - Subsystem name is the IMSID
  - Batch Backout utility
    - Subsystem name

SSYS Record Contents

- Subsystem name
- IRLM name
- Subsystem type
  - Batch
  - Online
  - Tracker
- Log open time stamp
- Status flags
- Authorized database names and status
  - the size of this part of the SSYS record varies
  - the number of databases or areas can grow large

SSYS Record Example

```
SSYS
 SSID=SYS3      LOG START=02.226 11:48:07.3
 SSTYPE=ONLINE  ABNORMAL TERM=OFF  RECOVERY STARTED=NO   BACKUP=NO
 TRACKED=NO     TRACKER TERM=OFF   SHARING COVERED DBS=NO
 IRLMID=**NULL**  IRLM STATUS=NORMAL        GSGNAME=IMSGSG1

 AUTHORIZED DATA BASES/AREAS=1      VERSION=8.1   XRF CAPABLE=NO
                                                   ENCODED
   -DBD-        -AREA-   -LEVEL-  -ACCESS INTENT-  -STATE-
   DEDBDD01     DD01AR0     1        UPDATE           6
```

IMS Batch Log Processing

- At Log Open, IMS calls DBRC with log information to record in the RECON
  - DBRC uses its Notify processing (DSPURT00)
    - Create PRILOG record
    - Create the LOGALL base record
    - Update the subsystem record
- At Log End of Volume (EOV), IMS calls DBRC with log information to write in the RECON
  - DBRC updates the PRILOG with volume information
- At Log Close, IMS calls DBRC with log information to record in the RECON
  - DBRC updates the PRILOG
- For dual logging, IMS makes similar calls to DBRC to write information about the secondary log in the RECON - the SECLOG record.

IMS Online Log Processing

- IMS Online Log open
  - Create PRIOLD record
  - Create SECOLD record
  - Create PRILOG and PRISLDS records
  - Create the LOGALL base record
  - Update the SSYS record
- IMS Online Log Switch
- IMS Online Log Close

PRILOG Record Contents

- Start time, Stop time
- Subsystem name (i.e. SSID)
- Log Version
- Global Service Group Name (GSGNAME)
- PRILOG Token
- Number of data sets (i.e. #DSN)
- First Record ID
- Each data set entry contains the following
  - Dataset name
  - Start, Stop time
  - First, Last Dataset Log Sequence Number
  - Unit, File, Volume, and Checkpoint information

PRILOG Record Example

```
PRILOG                                       RECORD SIZE=     304
 START = 02.231 10:28:00.2              *    SSID=SYS3     VERSION=8.1
 STOP   = 00.000 00:00:00.0                  #DSN=1
 GSGNAME=IMSGSG1
 FIRST RECORD ID= 000000000000027B       PRILOG TOKEN= 5
 EARLIEST CHECKPOINT = 02.231 10:28:03.7

 DSN=IMSVS.RLDSP.SYS3.D02231.T1028002.V01                 UNIT=SYSDA
 START = 02.231 10:28:00.2                FIRST DS LSN= 000000000000027B
 STOP   = 02.231 10:30:16.3               LAST  DS LSN= 0000000000000400
 FILE SEQ=0001    #VOLUMES=0001

   VOLSER=000000 STOPTIME = 02.231 10:30:16.3
     CKPTCT=2    CHKPT ID = 02.231 10:28:03.7
     LOCK SEQUENCE#=  000000000000
```

LOGALL Record Contents

- Start time of associated PRILOG
- Earliest allocation time stamp
- Number of Database datasets allocated
- DBD, DDN, and Alloc counts for each DBDS
  - This area of the LOGALL record varies in size depending on the number of DBDSs allocated.
  - Allocation count relates to the number of ALLOC records that exist in the RECON for a given DBDS for the log.

LOGALL Record Example

```
LOGALL
 START   = 02.231 10:28:00.2              *
 EARLIEST ALLOC TIME = 02.231 10:28:05.9
 DBDS ALLOC=1                        -DBD-    -DDN-    -ALLOC-
                                     DEDBDD01 DD01AR0   1
```

PRISLD Record Contents

- Start time, Stop time
- Subsystem name (i.e. SSID)
- Log Version
- Global Service Group Name (GSGNAME)
- PRILOG Token
- Number of data sets (i.e. #DSN)
- First Record ID
- Each data set entry contains the following
  - Dataset name
  - DS start, stop time
  - First, last DS Log Sequence Numbers
  - Unit, File, Volume, and Checkpoint information

PRISLD Record Example

```
PRISLD                                       RECORD SIZE=     304
 START = 02.231 10:28:00.2              *    SSID=SYS3     VERSION=8.1
 STOP  = 00.000 00:00:00.0                   #DSN=1
 GSGNAME=IMSGSG1
 FIRST RECORD ID= 000000000000027B       PRILOG TOKEN= 5

 DSN=IMSVS.SLDSP.SYS3.D02231.T1028002.V01                 UNIT=SYSDA
 START = 02.231 10:28:00.2                FIRST DS LSN= 000000000000027B
 STOP  = 02.231 10:30:16.3                LAST  DS LSN= 0000000000000400
 FILE SEQ=0001    #VOLUMES=0001
 CHECKPOINT TYPES=C0: SIMPLE=Y SNAPQ=Y DUMPQ=N PURGE=N FREEZE=N

   VOLSER=000000 STOPTIME = 02.231 10:30:16.3
     CKPTCT=2    CHKPT ID = 02.231 10:28:03.7
     LOCK SEQUENCE#=  000000000000
```

PRIOLD Record Contents

- Subsystem name
- Number of OLDS DD entries
- Earliest Checkpoint time stamp
- Data Set information
  - DD name
  - DSN
  - START and STOP times
  - First, Last DS LSN
  - Lock sequence number
  - Status
    - Archive status (e.g. ARC COMPLT, ARC STARTED,)
    - ARCHIVE JOB NAME
    - Available or Not
    - FEOV
  - PRILOG time

PRIOLD Record Example

```
PRIOLD
 SSID=SYS3            # DD ENTRIES=2
 EARLIEST CHECKPOINT = 02.231 10:28:03.7

 DDNAME=DFSOLP01   DSN=IMSTESTL.IMS01.OLDSP1
 START = 02.231 10:28:00.2                FIRST DS LSN= 000000000000027B
 STOP  = 02.231 10:30:16.3                LAST  DS LSN= 0000000000000400
 LOCK SEQUENCE# = 000000000000
 STATUS=ARC COMPLT                                FEOV=NO    AVAIL
 PRILOG TIME=02.231 10:28:00.2                ARCHIVE JOB NAME=JT103016
 VERSION=8.1

 DDNAME=DFSOLP00   DSN=IMSTESTL.IMS01.OLDSP0
 START = 02.231 10:30:16.3                FIRST DS LSN= 0000000000000401
 STOP  = 00.000 00:00:00.0                LAST  DS LSN= 0000000000000000
 LOCK SEQUENCE# = 000000000000
 STATUS=ACTIVE                                    FEOV=NO    AVAIL
 PRILOG TIME=02.231 10:28:00.2
 VERSION=8.1
```

Archive Processing (OLDS)

- GENJCL.ARCHIVE

- Archive Utility calls DBRC to:
  - Add SLDS entry to PRISLD/SECSLD records
  - Add RLDS entry to PRILOG/SECLOG records
    - SLDS entry added if RLDS not created
  - Set status of archived PRIOLD/SECOLD to ARC COMPLT
  - PRILOG closed when all OLDS for a subsystem are archived.

Archive Processing (SLDS or RLDS)

- Validate that SLDS or RLDS being archived exists in RECON
  - Match
    - DSN
    - file sequence number
    - volume serial numbers
  - Look for duplicates
  - Can take awhile

- At archive completion update the data set entry (DSE) with new
  - DSN
  - file sequence number
  - unit type
  - volume serial numbers

DB Authorization Processing

- Performed at initialization time for batch and first schedule for online

- Authorization for unregistered data bases granted unless FORCER specified
  - Database registered if DBD, DDN, and DSN match RECON

- General processing flow:
  - Check status in DB and DBDS records
  - Check current authorizations for compatibility with this one
  - If DBs requested can be granted, update SSYS and DB records

SSYS Record Example

```
SSYS
 SSID=SYS3      LOG START=02.022 15:49:42.8
 SSTYPE=ONLINE  ABNORMAL TERM=OFF  RECOVERY STARTED=NO   BACKUP=NO
 TRACKED=NO     TRACKER TERM=OFF   SHARING COVERED DBS=NO
 IRLMID=**NULL**  IRLM STATUS=NORMAL        GSGNAME=IMSGSG1

 AUTHORIZED DATA BASES/AREAS=1      VERSION=8.1   XRF CAPABLE=NO
                                                   ENCODED
   -DBD-        -AREA-   -LEVEL-  -ACCESS INTENT-  -STATE-
   DEDBDD01     DD01AR0     1        UPDATE           6
```

DBDS (Area) Record Example

```
 DBDS
  DBD=DEDBDD01  AREA=DD01AR0                                    TYPE=FP
  SHARE LEVEL=1             DSID=001 DBORG=DEDB   DSORG=VSAM
  GSGNAME=IMSGSG1              USID=0000000003
  AUTHORIZED USID=0000000003  RECEIVE USID=0000000003 HARD USID=0000000003
  RECEIVE NEEDED USID=0000000000
  CAGRP=**NULL**  GENMAX=2     IC AVAIL=0     IC USED=1     DSSN=00000002
  NOREUSE         RECOVPD=0      NOVSO  PREOPEN    NOPRELOAD
  CFSTR1=**NULL**         CFSTR2=**NULL**          NOLKASID
  DEFLTJCL=**NULL**  ICJCL=ICJCL    RECVJCL=ICRCVJCL  RECOVJCL=RECOVJCL
  DBRCVGRP=**NULL**

  FLAGS:                             COUNTERS:
    PROHIBIT AUTHORIZATION=OFF         AUTHORIZED SUBSYSTEMS   =1
                                       HELD AUTHORIZATION STATE=6
    IC NEEDED             =OFF         ADS AVAIL #             =1
    RECOV NEEDED          =OFF         REGISTERED ADS #        =1
    DATABASE LEVEL TRACK  =YES         EEQE COUNT              =0
    RECEIVE NEEDED        =OFF
    OFR REQUIRED          =NO
    TRACKING SUSPENDED    =NO
    HSSP CIC IN PROGRESS  =NO

  ADS LIST:
                                                                  CREATE
    -ADS DDN--ADS DSN-                                    -STAT- -RUNNING-
    DD01AR0  DD01AR0                                      AVAIL     NO

ASSOCIATED SUBSYSTEM INFORMATION:
                               ENCODED
     -SSID-   -ACCESS INTENT-  -STATE-  -SS ROLE-
    SYS3         UPDATE           6      ACTIVE
```

DBDS Update Processing

- Performed when first update to a DBDS occurs following allocation

- IMS calls DBRC to:
  - Check that DB is registered to DBRC
    - Match DBD, DDN, and DSN to RECON record
  - Create ALLOC record in RECON
  - Update LOGALL record
  - Return DSSN to IMS

ALLOC Record Contents

- ALLOC time stamp
- ALLOC Log Record ID (LRID)
- DEALLOC time stamp
- DEALLOC LRID
- Update Set ID (USID)
- Data Set Sequence Number (DSSN)
- PRILOG start time stamp

ALLOC Record Example

```
ALLOC
 ALLOC   =2003.190 11:18:26.4 -08:00     *  ALLOC LRID =0000000000000000
 DSSN=0000000004 USID=0000000005 START = 2003.190 11:16:52.6 -08:00
 DEALLOC =2003.190 11:18:52.6 -08:00      DEALLOC LRID =0000000000000000

ALLOC
 ALLOC   =2003.190 11:39:00.8 -08:00     *  ALLOC LRID =0000000000000000
 DSSN=0000000005 USID=0000000006 START = 2003.190 11:37:33.0 -08:00
```

DBDS Record Example

```
DBDS
 DSN=DBVHDJ05.CJXXD01E                                         TYPE=IMS
 DBD=DBVHDJ05  DDN=CJVHDG1E DSID=001 DBORG=HDAM   DSORG=VSAM
 CAGRP=CAGRP2    GENMAX=2     IC AVAIL=0     IC USED=2     DSSN=00000005
 NOREUSE         RECOVPD=0
 DEFLTJCL=**NULL**  ICJCL=ICJCL     OICJCL=OICJCL    RECOVJCL=RECOVJCL
 RECVJCL=ICRCVJCL
 FLAGS:                             COUNTERS:
   IC NEEDED      =OFF
   RECOV NEEDED   =OFF
   RECEIVE NEEDED =OFF                EEQE COUNT              =0
```

Database I/O Error Processing

- Performed for read and write errors

- Call DBRC to:

  - Update DB and DBDS records with Extended Error Queue Element (EEQE) information

  - Set "Recovery Needed" flag in DBDS record

  - Increment "Recovery Needed" counter in DB record

Database Reorganization

- Create and execute the DB Reorganizaton JCL
  - Do it yourself; no GENJCL support
- The IMS Reorganization process:
  - Calls DBRC upon completion to create the REORG record in the RECON

REORG Record Contents

- The REORG record is small and contains:
  - Time stamp when a reorganization occurred for the DBDS
    - RUN=
  - Update Set Identifier (USID)

- Actions that cause DBRC to create and write the record to the RECON:
  - NOTIFY.REORG
  - An IMS Utility
    - FP INITIALIZATION UTILITY (DBFUMIN0)
    - HD REORGANIZATION RELOAD (DFSURGL0)
    - REORG/LOAD, PREFIX UPDATE (DFSURGP0)
    - HISAM REORGANIZATION RELOAD (DFSURRL0)
  - Initial DB load (PROCOPT=L)

Image Copy Processing

- Generate the Image Copy JCL
  - Do it yourself
  - Let DBRC do it with GENJCL.IC command

- Execute generated IC JCL

- The IMS Image Copy utility
  - Calls DBRC to:
    - signon
    - validate the IC JCL
  - Does the image copy
  - Calls DBRC upon completion to
    - create the IC record and put it in the RECON
    - signoff

IC Record Contents

- Time stamps
  - Run - start time of the image copy processing
  - Stop - stop time or zeros
- Image Copy type
  - BATCH: Batch Image Copy
  - ONLINE: Online Image Copy (OIC)
  - CONCUR: Concurrent Image Copy (CIC)
  - SMSCIC
    - System Managed Storage (SMS) Concurrent Image Copy data sets: i.e. SMS IC with shared access
  - SMSNOCIC
    - System Managed Storage (SMS) Non-Concurrent Image Copy data sets: i.e. SMS IC with exclusive access
- Image Copy dataset information
- Record count - the number of records in the IC data set

IMAGE Record Example

```
IMAGE
 RUN     = 02.226 12:03:24.6              *  RECORD COUNT =33
 STOP    = 00.000 00:00:00.0                 BATCH      USID=0000000004

IC1
 DSN=IMSVS.DBVHDJ05.CJVHDG1E.IC.IC120322          FILE SEQ=0001
 UNIT=SYSDA                         VOLS DEF=0001 VOLS USED=0001
                                    VOLSER=222222



IMAGE
 RUN     = 02.226 12:04:27.3              *  RECORD COUNT =33
 STOP    = 02.226 12:04:27.6                 CONCUR     USID=0000000005

IC1
 DSN=IMSVS.DBVHDJ05.CJVHDG1E.IC.IC120424          FILE SEQ=0001
 UNIT=SYSDA                         VOLS DEF=0001 VOLS USED=0001
                                    VOLSER=222222
```

Change Accumulation Processing

- Generate the Change Accumulation JCL
  - Do it yourself
  - Let DBRC do it with GENJCL.CA command

- Execute generated CA JCL

- The IMS CA utility
  - Calls DBRC to validate the CA JCL
  - Processes the change accumulation
  - Calls DBRC upon completion to create the CA record and put it in the RECON

CA Record Contents

- CA Dataset name
- CA Group name
- Time stamps
  - Stop
  - Run
  - Purge
- Status flags
- Details about DBDs and DDNs processed

CA Record Example

```
CA
 DSN=IMSVS.CAGRP1.CA.CA170401                     FILE SEQ=1
 CAGRP=CAGRP1       STOP    = 02.226 09:36:08.4              *
                    UNIT=SYSDA      VOLS DEF=1    VOLS USED=1
                                    VOLSER=222222
 RUN     = 02.226 09:04:28.0
  DBD=DEDBJN21 DDN=DB21AR1  PURGETIME = 02.226 09:35:52.3
    CHANGES ACCUMULATED=YES COMPLETE CA=YES INDOUBT EEQES=NO
    LSN    = 000000000000      DSSN   = 0000000001
    LRID   = 0000000000000412  USID   = 0000000002
  DBD=DEDBJN21 DDN=DB21AR3  PURGETIME = 02.226 09:35:53.2
    CHANGES ACCUMULATED=YES COMPLETE CA=YES INDOUBT EEQES=NO
    LSN    = 000000000000      DSSN   = 0000000001
    LRID   = 000000000000042D  USID   = 0000000002
  DBD=DEDBJN21 DDN=DB21AR6  PURGETIME = 02.226 09:35:54.4
    CHANGES ACCUMULATED=YES COMPLETE CA=YES INDOUBT EEQES=NO
    LSN    = 000000000000      DSSN   = 0000000001
    LRID   = 0000000000000447  USID   = 0000000002
  DBD=DEDBJN21 DDN=DB21AR7  PURGETIME = 02.226 09:35:55.0
    CHANGES ACCUMULATED=YES COMPLETE CA=YES INDOUBT EEQES=NO
    LSN    = 000000000000      DSSN   = 0000000001
    LRID   = 000000000000045C  USID   = 0000000002
```

Database Recovery Processing

- Generate the Database Recovery JCL
  - Do it yourself
  - Let DBRC do it with GENJCL.RECOV command

- Execute generated RECOV JCL

- The IMS Database Recovery utility
  - Calls DBRC to validate the JCL
  - Processes the database recovery
  - Calls DBRC upon completion to create the RECOV record and put it in the RECON

RECOV Record Contents

- The RECOV record is small and contains:
  - The Database Recovery utility execution time for the DBDS
    - RUN=timestamp
  - The time to which the DBDS was recovered
    - RECOV TO= timestamp
  - The Update Set IDentifier, USID
    - RUN USID=
    - RECOV TO USID=
- Actions that cause DBRC to create and write the record to the RECON:
  - NOTIFY.RECOV
  - DB Recovery Utility

Backout Error Processing

- Invoked for:
  - Dynamic backout failures
  - /ERE backout failures
  - /ERE NOBMP
- Call DBRC to:
  - Increment "Backout Needed" counter in DB record

Batch Backout Processing

- Generate the IMS Batch Backout utility JCL
  - No GENJCL support

- Execute Batch Backout utility JCL

- The IMS Batch Backout utility
  - Calls DBRC (DSPUBK00) to validate the JCL
    - DBRC support introduced in IMS 4.1
    - Data in RECON used to validate input log(s)
  - Performs any backouts needed
  - Calls DBRC (DSPUBK00) upon completion to update the RECON

BACKOUT Record Contents

- Subsystem name (SSID)
- Number of units of recovery
- Recovery token describes a specific UOR
- Time stamp of the beginning of the UOR
  - found in the X'5607' log record
- PSB name associated withe the UOR
- UOR indicators
  - CANDIDATE, INFLT, INDT, BMP, ...
- Associated Data Bases
  - number of databases associated with the UOR
  - Information regarding each database
    - DBD
    - Backed Out (Yes or No)
    - Dynamic Backout failure (Yes or No)

BACKOUT Record Example

```
BACKOUT
 SSID=SYS3      #UORS=2

   RECOVERY TOKEN=E2E8E2F3404040400000000100000000
   TIME=04.060 22:38:01.8                PSB=PLVAPZ12
               INFLT   BMP   COLDEND
     ASSOCIATED DATA BASES=1

                  BACKED   DYN BKOUT
     -DBD-        -OUT -   -FAILURE-
     DIVNTZ02      NO          NO

   RECOVERY TOKEN=E2E8E2F3404040400000000200000000
   TIME=04.060 22:38:09.2                PSB=PSBEJK05
               INFLT   BMP   COLDEND
     ASSOCIATED DATA BASES=3

                  BACKED   DYN BKOUT
     -DBD-        -OUT -   -FAILURE-
     DBVHDJ05      NO          NO
     DBOHIDK5      NO          NO
     DXVHIDK5      NO          NO
```

DBRC Diagnostics

Problem Determination and Problem Source Indentification

- Messages
  - DSP0307I
  - DSP0300I
- Traces
  - DBRC Internal
  - External
- Tools
  - DSPURX00
  - AMASPZAP
  - AMS
    - IDCAMS
- DBRC Dumps
  - Important Control Blocks
    - DSPRSTG, DSPGDB, DFSBRLSB

Diagnostics: Messages

DBRC Messages have a 'DSP' prefix

- DSP0307I - issued by DBRC through IMS Version 6
  - Issued prior to ABENDU2480
  - Diagnosis of the problem very difficult without the dump
  - Many errcode-diagcode sets (about 10 pages)

```
DSP0307I INTERNAL DBRC ERROR
CODE errcode-diagcode
```

- DSP0300I - replaced DSP0307I in IMS V7
  - Issued prior to ABENDU2480
  - Diagnosis of the problem much improved and possible, in some cases, without the dump

```
DSP0300I INTERNAL DBRC ERROR
mmmmmmmm(aaaaaaaa) + X oooo # nn term/dump
DSP0300I   DIAG=dddddddd
```

Diagnostics: Traces

DBRC Internal

- Wrap around in-storage type
- Always active
- Indentified by the "TRACETBL" eyecatcher
- Trace Table size fixed by code
  - 4800 entries
  - Must ZAP module DSPTRACE to increase

DBRC External

- Normally inactive
- Activated by the User
  - Start GTF for USR-type records
  - CHANGE.RECON command with TRACEON
- Stopped by
  - CHANGE.RECON command with TRACEOFF
  - Stop GTF

Diagnostics: Tools

DSPURX00

- ► Use LIST commands to display RECON records of interest
  - LIST.RECON STATUS
  - LIST.HISTORY

AMASPZAP

- ► Display code
  - CSECT maintenance level in RESLIB
  - Determine offset of an instruction
- ► Modify an instruction
  - Force an ABEND at a certain point in the code

AMS

- ► IDCAMS
  - Print RECON records of interest

Diagnostics: AMASPZAP JCL

- DUMPT

```
//DUMPT    JOB NNNNN,MSGLEVEL=1
//**********************************************************************
//* JOB NAME:        DUMPT
//* JOB ENVIRONMENT: THIS JOB RUNS ON EC MACHINE 'IMSQA'
//* JOB DESCRIPTION: THIS JOB USES AMASPZAP TO DUMP ALL OF A LOAD
//*     MODULE OR JUST A CSECT OF IT FROM A PARTICULAR LIBRARY AS
//*     SPECIFIED IN THE SYSLIB JCL STATEMENT.
//*       MEMBER REFERS TO THE LOAD MODULE
//*       CSECT  REFERS TO A CSECT WITHIN THE LOAD MODULE
//*       ALL    WILL DUMP ALL CSECTS WITHIN THE LOAD MODULE
//**********************************************************************
/*ROUTE  PRINT
// EXEC PGM=AMASPZAP
//SYSPRINT DD SYSOUT=A
//SYSLIB  DD DSN=IMS.RESLIB,DISP=SHR
//SYSIN DD *
  DUMPT MEMBER ( CSECT | ALL )
/*
```

Diagnostics: Dumps

- DBRC Dumps
  - The DBRC internal trace table
    - Eyecatcher: TRACETBL
    - Wrap around type
    - Provides DBRC process flow details
    - Contains DFSBRLSC FUNC=, ID= codes
- Important DBRC Control Blocks
  - Router Storage: DSPRSTG
    - Contains table of DBRC parts and EPAs
    - Contains pointer to DSPGDB
    - Contains pointer to DBRC trace table
  - Global Data Block: DSPGDB
    - Pointer to DSPRSTG

Diagnostics: TRACETBL

Words 5-8 of the DBRC Trace Table ...

```
...   0554B480 0554FF80 0554C5C0 00000000      *TRACETBL.......-..........E{....*
...   0001BF30 0001E010 BAD9C6A7 08BEDD03      *DSPSTGET.....&........\..RFx....*
...   0001E010 0001E960 BAD9C6A7 08BF8183      *DSPSTGETe&.fe.....\...Z-.RFx..ac*
...   00000000 00000000 BAD9C6A7 08BFD483      *DSPURI00...8.............RFx..Mc*
...   D7C8E8E2 C9C3C1D3 40D6D7C5 D5404040      *MODIRCLE    O...PHYSICAL OPEN   *
...   40404040 40404040 00000000 00000000      *                        ........*
...   0001E960 0001F020 BAD9C6A7 08C22703      *DSPSTGETe.....i&..Z-..0..RFx.B..*
...   0001E960 0001F020 BAD9C6A7 08C32F03      *DSPSTFREe......8..Z-..0..RFx.C..*
...   0001E960 0001F020 BAD9C6A7 08C6DB03      *DSPSTGETe.........Z-..0..RFx.F..*
```

Notes:

Words 5, 6, and 7 of the TRACETBL entry are:

- First trace entry
- Last trace enty
- Next trace entry

Word 3 (not shown) of the DSPURI00 entry contains the address of DSPGDB

Diagnostics: TRACETBL in V8

```
ASID(X'002E') STORAGE  ------------------------------------------------
0BA7B700   E3D9C1C3   C5E3C2D3   00025900   000006C6    | TRACETBL.......F |
0BA7B710   0BA7B720   0BAA0F20   0BA90E60   0BAA0FA0    | .x.......z.-.... |
0BA7B720   606EF1E3   C9D4C5F0   80013528   0BA35080    | ->1TIME0.....t&. |
0BA7B730   00006620   00016010   02190F22   20185621    | ......-......... |
0BA7B740   F04C60E3   C9D4C5F0   00013528   000010F4    | 0<-TIME0.......4 |
0BA7B750   00006620   00016010   02190F22   20185622    | ......-......... |
0BA7B760   606EF1E4   D9C9F0F0   8001369C   0BA49088    | ->1URI00.....u.h |
0BA7B770   00006620   00016010   02190F22   20185622    | ......-......... |
0BA7B780   C4E2D7E4   D9C9F0F0   00014748   00000000    | DSPURI00........ |
0BA7B790   00000000   00000000   02190F22   20185622    | ................ |
0BA7B7A0   D4D6C4C9   D9C3C1D9   40404040   D6000600    | MODIRCAR    O... |
0BA7B7B0   D7C8E8E2   C9C3C1D3   40D6D7C5   D5404040    | PHYSICAL OPEN    |
0BA7B7C0.:0BA7B7CF.--All bytes contain X'40', C' '
0BA7B7D0   40404040   40404040   00000000   00000000    |         ........ |
0BA7B7E0   606EF2E4   C3D7F4F0   00000A7A   00010750    | ->2UCP40...:...& |
0BA7B7F0   00016010   00016960   02190F22   20185623    | ..-.....-....... |
0BA7B800   F14C60E4   C3D7F4F0   0BA49B02   00000B52    | 1<-UCP40.u...... |
0BA7B810   00016010   00016960   02190F22   20185623    | ..-.....-....... |
0BA7B820   606EF2E4   D9C9F1F0   00000F86   0BA4E440    | ->2URI10...f.uU  |
0BA7B830   00016010   00016960   02190F22   20185624    | ..-.....-....... |
0BA7B840   606EF3E4   D9C9F2F0   00000208   0BA503D8    | ->3URI20.....v.Q |
0BA7B850   00016960   00016D70   02190F22   20185624    | ...-.._......... |
```

Diagnostics: DSPRSTG

```
00000D68 05500F48 C0E00000 00000001   *        DSPRSTG .....&..{\......*
00000000 00000000 00000000 00000000   *.......8........-...............*
00000000 00000000 00000000 00000000   *................................*
00000000 00000000 C4E2D7E4 D9E2E2E3   *........................DSPURSST*
00005DE0 00005E50 00000000 00000000   *...<..........)...)\..;&........*
00000000 00000000 00000000 00000000   *................................*
00000000 00000000 00000000 00000000   *................................*
C4E2D7C1 C4E2F0F0 80000001 05538440   *................DSPADS00......d *
C4E2D7C1 D3C4F0F0 80000003 05538C88   *DSPADTIM.......}DSPALD00.......h*
50E050E0 58F050E4 05EF0000  00005130  *DSPAMS00.......\&\&\.0&U........*
C4E2D7C3 C1C2D5F0 80000007 0001A1A8   *&\&\.0&U....... DSPCABN0.......y*
C4E2D7C3 C8D2E6C4 80000009 05549F78   *DSPCEXT0.......hDSPCHKWD........*
C4E2D7C4 C5D8C540 8000000B 0554A4D0   *DSPDEQ00........DSPDEQE ......u}*
C4E2D7C4 D3E3F0F0 8000000D 0553B860   *&\&\.0&U........DSPDLT00.......-*
50E050E0 58F050E4 05EF0000 000051D0   *DSPDTM  ......f.&\&\.0&U.......}*
C4E2D7C8 C9C3C2C7 80000011 0553A640   *DSPHIC00......tQDSPHICBG......w *
C4E2D7C9 C3D7F0F0 80000013 0553E510   *DSPHICED........DSPICP00......V.*
C4E2D7D1 C2E2C1D3 00000015 055DE508   *DSPJBMAI.....)..DSPJBSAL.....)V.*
C4E2D7D1 C2E2C4C2 00000017 055E09F0   *DSPJBSCA.....)6QDSPJBSDB.....;.0*
C4E2D7D1 C2E2C9C3 00000019 055E1AA0   *DSPJBSEL.....)..DSPJBSIC.....;..*
C4E2D7D1 C2E2D9D3 0000001B 055E5388   *DSPJBSOL.....; -DSPJBSRL.....;.h*
```

RECON Record Types: Table 1

Common Name	Type	DSECT	List Name	Comment
RECON Header	X'01'	DSPRCNRC	RECON	
RECON Header Extension	X'01'	DSPRCR1	n/a	Not listed.
RECON Audit Record	X'02'	DSPMUPHD	n/a	Not listed; rarely seen.
PRILOG	X'05'	DSPLOGRC	PRILOG	
Interim PRILOG	X'06'	DSPLOGRC	IPRI	Seldom seen
LOGALL	X'07'	DSPLGARC	LOGALL	
SECLOG	X'09'	DSPLOGRC	SECLOG	
Interim SECLOG	X'0A'	DSPLOGRC	ISEC	Seldom seen
Change Accum Group	X'0F'	DSPCAGRC	CAGRP	
Change Accum Execution	X'11'	DSPCHGRC	CA	

RECON Record Types: Table 2

Common Name	Type	DSECT	List Name	Comment
Available CA	X'51'	DSPCHGRC	CA	
DBDS Group	X'16'	DSPDGRC	DBDSGRP, DBGRP, RECOVGRP	
Data Base Header	X'18'	DSPDBHRC	DB	
DB Partition	X'19'	DSPPTNRC	DB	Listed in combination with the Part DB rcd
Data Base Data Set	X'20'	DSPDSHRC	DBDS	
Area Recov	X'20'	DSPDSHRC	DBDS	Listed in combination with the Area Auth
Area Auth	X'21'	DSPDBHRC	DBDS	Listed in combination with the Area Recov
ALLOC	X'28'	DSPALLRC	ALLOC	
Image Copy	X'2D'	DSPIMGRC	IMAGE	
Reorg	X'32'	DSPRRGRC	REORG	

RECON Record Types: Table 3

Common Name	Type	DSECT	List Name	Comment
Recovery	X'37'	DSPRCVRC	RECOV	
Available IC	X'6D'	DSPIMGRC	IMAGE	
Backout	X'35'	DSPBKORC	BACKOUT	
Global Service Group	X'3A'	DSPGSGRC	GSG	
Tracking Subsystem	X'3E'	DSPSSRC	SSYS	RSR
Subsystem	X'3F'	DSPSSRC	SSYS	
Primary SLDS	X'43'	DSPSLDRC	PRISLD	
Tracking Primary SLDS	X'44'	DSPSLDRC	PRITSLD	RSR
Interim PRISLD	X'45'	DSPSLDRC	IPRISL	Seldom seen
Interim Tracking Primary SLDS	X'46'	DSPSLDRC	IPRITSLD	Seldom seen (RSR)

RECON Record Types: Table 4

Common Name	Type	DSECT	List Name	Comment
Secondary SLDS	X'47'	DSPSLDRC	SECSLD	
Tracking Secondary SLDS	X'48'	DSPSLDRC	SECTSLD	Seldom seen (RSR)
Interim Secondary SLDS	X'49'	DSPSLDRC	ISECSL	Seldom seen
Interim Tracking Secondary SLDS	X'50'	DSPSLDRC	ISECTSLD	Seldom seen (RSR)
Primary OLDS	X'53'	DSPOLDRC	PRIOLD	
Interim Primary OLDS	X'55'	DSPOLDRC	IPRIOL	Seldom seen
Secondary OLDS	X'57'	DSPOLDRC	SECOLD	
Interim Primary OLDS	X'59'	DSPOLDRC	ISECOL	Seldom seen

DBRC V8 Enhancements Summary

- 16 MB RECON records
- PRILOG Compression
- Automatic RECON Loss Notification
- DBRC Command Authorization
- Minimum Version Specification
- DEDB Enhancements

V8: 16 MB RECON records

- Maximum size of 'logical' RECON record is 16 megabytes
  - Eliminates potential outages due to RECON records reaching maximum VSAM record size
    - PRILOG, DBDS, SUBSYS
- DBRC implements its own RECON record spanning
  - RECON records are segmented into multiple VSAM records

V8: PRILOG Compression

- In previous releases compression is attempted:
  - During archive when threshold reached - 50% or 75%
  - When DELETE.LOG INACTIVE is issued

- In IMS V8:
  - On every archive
  - DELETE.LOG INACTIVE
  - If no DS entries compressed:

```
DSP1150I LOG RECORD(S) COULD NOT BE
          COMPRESSED, RECORD TIME = timestamp
          reason type = timestamp
```

    - Reason types:
      - EARLIEST ALLOC TIME
      - LOG RETENTION TIME
      - EARLIEST CHECK POINT

V8: Automatic RECON Loss Notification

- RECON reconfiguration with previous IMS Releases
  - When a DBRC instance detects bad RECON, it begins the reconfiguration process
    - Copies good RECON to spare
    - Deallocates the bad RECON
  - Bad RECON must be deleted and redefined to create "new" spare
    - Deletion requires deallocation of "bad" RECON by each subsystem (i.e. DBRC instance)
      - Requires each DBRC to access RECON

- Automatic RECON Loss Notification (ARLN)
  - Option in IMS V8 to make RECON reconfiguration with other IMS subsystems automatic
  - DBRC instance beginning reconfiguration notifies other DBRCs via SCI
  - Other DBRCs invoke reconfiguration process immediately
  - Eliminates wait for next access to RECONs

V8: DBRC Command Authorization

- Allows the installation to control the use of DBRC commands
  - Commands can be authorized at the "command verb" level, the "resource type" level or the "resource" level
    - DELETE
    - CHANGE.DB
    - CHANGE.DB DBD(AAA)
- Supported Environments:
  - DBRC Utility (DSPURX00)
  - HALDB Partition Definition Utility
- Does not support /RMxxxxxx commands

V8: Minimum Version Specification

- Used to specify the lowest version of IMS allowed to Signon
  - INIT.RECON   MINVERS(**<u>61</u>**|71|81)
    - Default is IMS 6.1
  - CHANGE.RECON MINVERS(**<u>61</u>**|71|81)
    - Signon denied for IMS subsystems at lower release levels

- IMS line item requirements
  - APPC/OTMA Shared Queues - IMS 8.1

- Time History Table deleted in V8
  - Used to support IMS V5 format timestamps
  - COEX and NOCOEX keywords ignored in INIT.RECON and CHANGE.RECON commands
  - THT and REPTHT keywords are ignored with CHANGE.RECON command

V8: DEDB Enhancements

- Nonrecoverable DEDBs
  - Useful for scatchpad, work, or temporary databases
- Greater than 240 areas
  - The upper limit of DEDB areas has been increased from 240 to 2048.

Further Information

- IMS website: http://www-3.ibm.com/software/data/ims/
  - IMS Publications Library
    - DBRC Guide and Reference
    - IMS Release Planning Guide
  - Presentations
  - Redbooks
    - Database Recovery Control (DBRC) Examples and Usage Hints (SG24-3333-01)

RETURN TO INDEX