

CICS Transaction Server for z/OS



CICSplex SM Concepts and Planning

Version 3 Release 1

CICS Transaction Server for z/OS



CICSplex SM Concepts and Planning

Version 3 Release 1

Note!

Before using this information and the product it supports, be sure to read the general information under "Notices" on page 139.

Fourth edition (July 2010)

This edition applies to Version 3 Release 1 of CICS Transaction Server for z/OS, program number 5655-M15, 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 the product.

© Copyright IBM Corporation 1994, 2010.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

| | |
|--|------|
| Preface | vii |
| Who this book is for | vii |
| What you need to know | vii |
| Notes on terminology. | vii |
| Summary of changes | ix |
| Changes made to this book for CICS Transaction Server for z/OS, Version 3 | |
| Release 1 | ix |
| Changes made to this book for CICS Transaction Server for z/OS, Version 2 | |
| Release 3 | ix |
| Changes made to this book for CICS Transaction Server for z/OS, Version 2 | |
| Release 2 | ix |
| New and changed function in CICS Transaction Server for OS/390 Release 3 | x |
| Dynamic routing | x |
| CICS business transaction services | xi |
| Temporary storage queue models | xi |
| Long temporary storage queue names | xii |
| Resource Recovery Services (RRS) | xii |
| Sysplex enqueue models | xii |
| Support of IIOp inbound to Java applications. | xiii |
| Coupling facility data tables | xiii |
| Support for CICS Web enhancements | xiii |
| Java Virtual Machine | xiv |
| Open Transaction Environment | xiv |
| FEPI | xiv |
| Support for Tivoli Global Enterprise Manager. | xiv |
| Changes made to this book for CICS Transaction Server for OS/390 Release 3 | xv |
| Chapter 1. Introducing CICSplex SM | 1 |
| CICS system connectivity | 2 |
| Features of CICSplex SM | 2 |
| Where next? | 6 |
| Chapter 2. The CICSplex SM environment | 7 |
| Overview | 7 |
| The CICSplex. | 9 |
| The managed application system (MAS) | 9 |
| CICSplex SM address space (CMAS) | 10 |
| Environment Services System Services (ESSS) | 11 |
| The CICSplex SM Web User Interface | 11 |
| The coordinating address space (CAS) | 12 |
| The CICSplex SM end user interface (EUI) | 12 |
| CICSplex SM objects | 13 |
| The data repository | 15 |
| The batched repository-update facility | 15 |
| Support for Tivoli Business Systems Manager | 15 |
| Where next?. | 16 |
| Chapter 3. Designing your CICSplex SM environment | 17 |
| Designing your CICSplexes | 17 |
| Identifying your CICS systems and regions | 17 |
| Which CICS systems can be managed by CICSplex SM? | 19 |
| How many CICSplexes are there in your enterprise? | 19 |

| | |
|---|-----------|
| Identifying system groups | 21 |
| Locating CMASs | 24 |
| Where to install CMASs | 24 |
| CMAS-to-CMAS links | 25 |
| CMAS-to-CMAS links for multiple CICSplexes | 28 |
| Locating the maintenance point CMAS | 28 |
| Locating CASs | 30 |
| Where CASs are required | 30 |
| CAS-to-CAS links | 31 |
| Naming your CICSplex SM entities | 33 |
| A staged implementation | 34 |
| Updating the initial design | 34 |
| Where next? | 34 |
| | |
| Chapter 4. Defining the CICSplex SM configuration and topology | 35 |
| Defining your CICSplex SM configuration | 35 |
| Defining CAS-to-CAS links | 35 |
| Defining the CMAS configuration | 35 |
| Defining CICSplex topology | 37 |
| Preparing CICSplex definitions | 37 |
| Preparing CICS system definitions | 37 |
| Where next? | 40 |
| | |
| Chapter 5. Managing resources using Business Application Services (BAS) 41 | 41 |
| What is Business Application Services (BAS)? | 41 |
| Limitations of RDO | 41 |
| BAS concepts | 41 |
| Two forms of BAS | 43 |
| Advantages of using BAS | 45 |
| Managing BAS | 45 |
| Migrating your resource definitions | 48 |
| Defining resources | 49 |
| Validating resources | 49 |
| Defining resource groups | 50 |
| Using migration form BAS | 50 |
| Using full-function form BAS | 50 |
| Using resource assignments | 51 |
| Using resource descriptions | 51 |
| Installing applications and resources | 52 |
| Security considerations | 52 |
| Planning for BAS | 53 |
| Implementing BAS | 53 |
| Creating migration form BAS | 54 |
| Migrating to the use of resource assignments | 54 |
| A recommended approach | 54 |
| Where next? | 55 |
| | |
| Chapter 6. Managing workloads | 57 |
| What is workload management? | 57 |
| What is dynamic routing? | 57 |
| Dynamic routing models | 58 |
| WLM functions | 60 |
| Advantages of WLM | 61 |
| Workload balancing | 61 |
| The queue algorithm | 62 |
| The goal algorithm | 62 |

#

| | |
|--|-----------|
| Cross CEC routing | 63 |
| Workload separation | 63 |
| Intertransaction affinity | 64 |
| How intertransaction affinities are honored | 64 |
| Managing workloads | 65 |
| Planning for WLM | 67 |
| Workload balancing or workload separation? | 67 |
| Identifying the workloads | 68 |
| Identifying intertransaction affinities | 69 |
| Implementing WLM | 69 |
| Identifying dynamic routing candidates | 69 |
| Implementing workload balancing | 70 |
| Adding workload separation requirements | 70 |
| Defining intertransaction affinities | 71 |
| A recommended approach. | 71 |
| Where next? | 72 |
| Chapter 7. Exception reporting using real-time analysis (RTA) | 73 |
| What is real-time analysis? | 73 |
| Why automate? | 73 |
| The RTA external notifications | 74 |
| RTA functions | 74 |
| Advantages of RTA | 74 |
| System Availability Monitoring (SAM) | 75 |
| Customizing the SAM external notifications | 75 |
| Automation and SAM | 76 |
| MAS resource monitoring (MRM) | 76 |
| Automation and MRM | 77 |
| Analysis point monitoring (APM) | 77 |
| Managing RTA | 77 |
| Planning for RTA | 79 |
| Implementing RTA. | 80 |
| Implementing system availability monitoring (SAM). | 80 |
| Implementing MAS resource monitoring (MRM) | 80 |
| Implementing analysis point monitoring (APM) | 81 |
| A recommended approach. | 81 |
| Where next? | 82 |
| Chapter 8. Collecting statistics using CICSplex SM monitoring | 83 |
| What is monitoring? | 83 |
| Monitoring functions | 83 |
| Advantages of monitoring | 83 |
| Monitoring of resources. | 83 |
| The monitor interval | 84 |
| The sample interval | 84 |
| Managing monitoring functions | 84 |
| Planning CICSplex SM monitoring | 86 |
| Implementing CICSplex SM monitoring. | 88 |
| Activating the resource status facility | 89 |
| Using the CICS Monitoring Facility (CMF) | 89 |
| A recommended approach. | 89 |
| Where next? | 90 |
| Chapter 9. Planning CICSplex SM installation and setup | 91 |
| Software, hardware, and storage requirements | 91 |
| Prerequisite software release levels | 91 |

| | |
|---|------------|
| Prerequisite hardware | 92 |
| Storage requirements | 92 |
| CICSplex SM naming conventions | 92 |
| Generic names | 93 |
| Security planning | 93 |
| Protecting access to CICSplex SM functions and CICS resources | 94 |
| Special considerations for BAS | 94 |
| CICS command and resource checking | 94 |
| Defining time zones | 94 |
| An overview of the installation process | 95 |
| Preparing to set up a CAS | 95 |
| Preparing to set up a CMAS | 96 |
| Using the batched repository-update facility | 97 |
| Preparing to set up a MAS | 98 |
| Reuse of CICSplex SM object definitions | 98 |
| Where next? | 99 |
| | |
| Chapter 10. The CICSplex SM Starter Set | 101 |
| Why is the Starter Set provided? | 101 |
| The Starter Set configuration | 101 |
| The Starter Set installation requirements | 102 |
| CICSplex SM definitions supplied with the Starter Set | 104 |
| Sample CAS definitions | 104 |
| Sample configuration definitions | 105 |
| Sample topology definitions | 105 |
| BAS definitions | 107 |
| Sample Workload definitions | 111 |
| Sample Monitor definitions | 113 |
| Sample Analysis definitions | 116 |
| The Starter Set naming convention | 125 |
| | |
| Bibliography | 127 |
| The CICS Transaction Server for z/OS library | 127 |
| The entitlement set | 127 |
| PDF-only books | 127 |
| Other CICS books | 129 |
| Determining if a publication is current | 129 |
| | |
| Accessibility | 131 |
| | |
| Index | 133 |
| | |
| Notices | 139 |
| Trademarks. | 140 |
| | |
| Sending your comments to IBM | 141 |

Preface

This book provides a general introduction to the CICSplex[®] System Manager (CICSplex SM) element of CICS[®] Transaction Server for z/OS[®], Version 3 Release 1.

- It describes, at a high level, how CICSplex SM works, and introduces some new concepts (see Chapter 1, “Introducing CICSplex SM,” on page 1).
- It introduces the main functions and benefits of CICSplex SM as a CICS system management tool (see Chapter 2, “The CICSplex SM environment,” on page 7).
- It explains how to approach the design of a CICSplex SM configuration (Chapter 3, “Designing your CICSplex SM environment,” on page 17).
- It describes how to start defining your CICSplex SM environment (see Chapter 4, “Defining the CICSplex SM configuration and topology,” on page 35).
- It describes how to plan for use of CICSplex SM’s Business Application Services, workload management, real-time analysis, and monitoring functions. See Chapter 5, “Managing resources using Business Application Services (BAS),” on page 41, Chapter 6, “Managing workloads,” on page 57, Chapter 7, “Exception reporting using real-time analysis (RTA),” on page 73, and Chapter 8, “Collecting statistics using CICSplex SM monitoring,” on page 83, as appropriate.
- It describes how to prepare for the installation of CICSplex SM (see Chapter 9, “Planning CICSplex SM installation and setup,” on page 91).
- It describes the CICSplex SM Starter Set (see Chapter 10, “The CICSplex SM Starter Set,” on page 101).

Who this book is for

This book is intended for anyone who needs a general introduction to CICSplex SM, and for system designers, planners, and programmers responsible for preparing an enterprise for the introduction of CICSplex SM.

What you need to know

Readers of this book are assumed to be familiar with CICS terminology and operation.

Notes on terminology

Unless otherwise indicated, in the text of this book:

- The term **CICSplex SM** (spelled with an uppercase letter ‘P’) means the IBM[®] CICSplex System Manager element of CICS Transaction Server. The term **CICSplex** (spelled with a lowercase letter ‘p’) means the largest set of CICS systems to be managed by CICSplex SM as a single entity.
- The term **CICS system** refers to a single administration unit of CICS on any of the platforms managed by CICSplex SM. On MVS[™], this term is synonymous with **CICS region**.
- The term **MVS** refers to the MVS/Enterprise Systems Architecture (MVS/ESA) operating system.

preface

Summary of changes

This book is based on the CICS Transaction Server for z/OS, Version 2 Release 3 edition of *CICSplex System Manager Concepts and Planning*. It has been updated to incorporate changes made for CICS Transaction Server for z/OS, Version 3 Release 1.

Changes made since the last edition are marked by vertical bars in the left margin.

Changes made to this book for CICS Transaction Server for z/OS, Version 3 Release 1

CICSplex SM support for the CICS for Windows component of IBM TXSeries (also known as NT 4.3 and NT 5.0) is no longer provided in CICS Transaction Server for z/OS, Version 3 Release 1. Therefore, it is no longer possible to set up a CICSplex SM remote MAS agent for Windows.

However, customers, who wish to do so, can continue to use the CICS Transaction Servers Version 2.3 or Version 2.2 for CICSplex SM support of TXSeries.

Changes made to this book for CICS Transaction Server for z/OS, Version 2 Release 3

CICSplex SM support for Link3270 bridge workload management is introduced in this release. It is covered in Chapter 6, “Managing workloads,” on page 57.

CICS for OS/2 is not supported as an RMAS in CICS Transaction Server for z/OS, Version 2 Release 3.

A CICSplex SM CMAS will now run only on a CICS system at the same release level. The ability to run the latest release of CICSplex SM on older releases of CICS in the CMAS has been removed. Similarly, a MAS being used as a CICS Transaction Server for z/OS, Version 2 Release 3 CICSplex SM Web User Interface server must also be at the same CICS release level as the Web User Interface.

Changes made to this book for CICS Transaction Server for z/OS, Version 2 Release 2

The following change has been made to this book:

- Information on CICSplex SM support for directly or indirectly connectable CICS/VSE systems from CICS Transaction Server for z/OS, Version 3 Release 1 is removed.
- Information on CICSplex SM support for Enterprise JavaBean technology.

There has been a change in CICSplex SM field naming conventions in this release. Data set name fields such as DSNAME, file name fields such as LOCFILE and REMFILE, and transient data queue name fields such as EXTRATDQ and INTRATDQ are now case-sensitive. When entering data set and file names into the CICSplex SM interfaces (end user interface, API and the web user interface), ensure that you enter the data in the correct case. In previous releases of CICSplex SM, the data set names and file names are automatically converted to upper case.

New and changed function in CICS Transaction Server for OS/390 Release 3

The following additions and changes have been made to the functions of the CICSplex SM component of CICS Transaction Server for OS/390 Release 3.

Dynamic routing

CICS Transaction Server for OS/390 Release 3 introduces extensions to the CICSplex SM dynamic routing program (called the dynamic *transaction* routing program in previous releases). In addition to dynamic routing of transactions initiated at a terminal, you can now dynamically route:

- Transactions invoked using the EXEC CICS START commands that are associated with a terminal
- Transactions invoked using the EXEC CICS START commands that are not associated with a terminal
- Distributed program links, including:
 - The CICS Web support
 - The CICS Transaction Gateway
 - EXCI calls
 - CICS Client ECI calls
 - Distributed Computing Environment (DCE) remote procedure calls (RPCs)
 - Open Network Computing (ONC) remote procedure calls (RPCs)
 - Internet Inter-Object Request Block Protocol (IIOP)
 - Any function that issues an EXEC CICS LINK PROGRAM request
- Transactions associated with CICS business transaction services (BTS) activities (see “CICS business transaction services” on page xi)

These new functions allow you to integrate workload management for requests from all these sources and thereby improve performance and workload throughput. You specify which transactions and programs may be dynamically routed, and the CICS regions in which they may be run.

The new dynamic functions are supported by:

- A new workload management view, WLMWAOS, which shows summarized information about all target regions that are associated with a workload that is within the CICSplex identified as the context. WLMWAOS is a summary form of the WLMWAOR view.
- A new workload management view, WLMAWTOS, which shows summarized information about all routing regions that are associated with a workload that is within the CICSplex identified as the context. WLMAWTOS is a summary form of the WLMAWTOR view.
- A new field, Dynamic, on the BAS PROGDEF view, which specifies whether or not an EXEC CICS LINK to the named program may invoke dynamic routing.
- A new field, Routable, on the BAS TRANDEF view, which specifies whether or not an eligible EXEC CICS START TERMID should be shipped to the terminal-owning region (TOR) for potential dynamic routing.

An overview of dynamic routing is given in “What is workload management?” on page 57. Full details are given in *CICSplex System Manager Managing Workloads*.

CICS business transaction services

CICS business transaction services (BTS) extends the CICS API and provides support services that make it easier to model complex business transactions. Using BTS, each action that makes up the business transaction is implemented as one or more CICS transactions, in the traditional manner. However, a top-level program is used to control the overall progress of the business transaction. The top-level program manages the inter-relationship, ordering parallel execution, commit scope, recovery, and restart of the actions that make up the business transaction. For full details about BTS, see *CICS Business Transaction Services*.

BTS processes are CICSplex-enabled, and take full advantage of CICSplex SM's workload separation and workload balancing functions. The CICS system group within a CICSplex, across which related BTS processes and activities may run, is known as a **BTS-set**. When an activity is attached, WLM routes the activity to the most appropriate region in the CICSplex. You can separate a BTS workload based on **process-type**. Workload management and the CICS Transaction Affinities Utility understand affinities between BTS processes and activities.

CICSplex SM support for BTS is provided by:

- A new BAS resource definition object, PROCDEF, which you use to define BTS process types. All other BTS resources are created dynamically when you specify a name in your application.
- New operations views:
 - PROCTYP, a general view of all installed process types
 - PROCTYPD, a detailed view of a selected process type
 - PROCTYPS, a summary view of all installed process types.
- Workload management changes:
 - A new view, WLMATAFD, which provides a detailed view of the properties of a single active BTS affinity. You can use this view to display the contents of the BTS affinity key in hexadecimal format.
 - The TRANDEF and WLMSPEC views have been enhanced to allow you to specify the BTS BAPPL affinity relation, and the BTS ACTIVITY and PROCESS affinity lifetimes.

Temporary storage queue models

CICS Transaction Server for OS/390 Release 3 introduces Resource Definition Online (RDO) support for the Temporary Storage Table (TST). For CICSplex SM, support for this function is provided by:

- A new BAS resource definition object TSMDEF, which allows you to specify a temporary storage queue name prefix and associate attributes with that name.
- New operations views:
 - TSMODEL, a general view of all currently available temporary storage queue models.
 - TSMODELD, a detailed view of a temporary storage model.
 - TSMODELS, a summary view of temporary storage models.
 - TSPOOL, a general view of temporary storage shared pools.
 - TSQSHR, a general view of shared temporary storage queues.
 - TSQSHRD, a detailed view of a shared temporary storage queue.
 - TSQSHRS, a summary view of shared temporary storage queues.

The existing temporary storage operations views, TSQ, TSQS, TSQGBL, and TSQGBLS, remain unchanged. However, you can now delete temporary storage queues from the TSQ, TSQSHR and TSQNAME views by entering the command DEL. A new TSQ Deletion Panel asks you to confirm the deletion.

Long temporary storage queue names

Users have found it difficult to generate temporary storage queue names that are unique. Typically, queue names were generated from the transaction id and the terminal id but unique names could be guaranteed only where the terminal id was unique and the transaction required one queue only. Many users like to have the user id as part of the queue name, so that the queue data is tied to the user rather than the terminal, but then there is no room for the transaction id which is required if properties, for example, recoverability and security, are to be associated with the queue via the temporary storage table (TST).

To provide greater flexibility, the length of temporary storage queue names has been increased to a maximum of 16 characters. For example, users can generate queue names of the form:

XXXXaUUUUUUUU

where:

XXXX is the transaction id.

a is a suffix allowing the transaction or user to have more than one queue.

UUUUUUUU

is the user id.

The TST or temporary storage model prefix can then be XXXX or XXXXa.

For CICSplex SM, support for this function is provided by:

- New operations views:
 - TSQNAME, a general view of all non-shared temporary storage queues.
 - TSQNAMED, a detailed view of a non-shared temporary storage queue.
 - TSQ NAMES, a summary view of all non-shared temporary storage queues.

Resource Recovery Services (RRS)

CICS Transaction Server for OS/390® Release 3 allows the use of Recoverable Resources Management Services (RRMS) facilities in an MVS image.

CICSplex SM supports this function through:

- A new value of WAITRRMS for the Wait Cause field of the UOWORKD view.
- A new Protocol field on the UOWLINKD view, that may take values of RRMS or blank.
- A new RRMS Status field on the CICSRGND view, that may take the values OPEN, CLOSED, or N/A.

Sysplex enqueue models

CICS Transaction Server for OS/390 Release 3 extends the CICS application programming interface to serialize resources across a set of CICS systems. The set of CICS systems defined as forming the resource set is called the *enqueue scope*.

CICSplex SM supports this function through:

- A new BAS resource definition ENQMDEF, which defines a global enqueue model.

- New operations views:
 - ENQMDL, a general view of enqueue models.
 - ENQMDLD, a detailed view of a selected enqueue model.
 - ENQMDLS, a summary view of enqueue models.
- A new field, Enqscope, on the UOWENQ view.

Support of IOP inbound to Java applications

CICSplex SM supports IOP inbound to Java™ applications through:

- A new BAS resource definition RQMDEF, which defines a CICS request model resource.
- New operations views:
 - RQMODEL, a general view of request models and their associated transactions.
 - RQMODEL, a detailed view of a selected request model.
 - RQMODELS, a summary view of all installed request models.

Coupling facility data tables

CICS Transaction Server for OS/390 Release 3 introduces support for coupling facility data tables. Part of this function is the implementation of new attributes on the FILE resource definition to specify a coupling facility data table pool to be associated with the file. In support of this function, CICSplex SM support of the file inquiry facility is extended by:

- New operations views:
 - CFDTPOOL, a general view of coupling facility data table pools associated with the file.
 - CFDTPOOD, a detailed view of a selected coupling facility data table pool.
 - CFDTPOOS, a summary view of coupling facility data table pools.
- The existing file operations view, CMDT, and its associated detail view, CMDTD, and summary view, CMDTS, have been amended to reflect support for coupling facility data tables.
- Two new detail views for CMDT:
 - CMDT2, for detailed information relating to a CICS- or user-maintained data table, or a coupling facility data table. You can hyperlink to this view from the Table Info field of the CMDTD view.
 - CMDT3, for statistical information relating to a data table file. You can hyperlink to this view from the Data Set Info field of the CMDT2 view.
- The FILE operations view is amended to reflect support of coupling facility data tables.

Support for CICS Web enhancements

CICS Transaction Server for OS/390 Release 3 introduces various enhancements to the CICS Web support. Part of this function is the introduction of new CICS RDO resource definitions, HTMLTEMPLATE and TCPIPSERVICE. CICSplex SM supports this function by:

- New BAS views:
 - DOCDEF, which defines document templates.
 - TCPDEF, which defines TCP/IP services.
- New operations views for document templates:
 - DOCTEMP, a general view of document templates.

- DOCTEMPD, a detailed view of a selected document template.
- DOCTEMPS, a summary view of document templates.
- New operations views for TCP/IP services:
 - TCPIPS, a general view of TCP/IP services.
 - TCPIPSD, a detailed view of a selected TCP/IP service.
 - TCPIPSS, a summary view of TCP/IP services.

Java Virtual Machine

CICS Transaction Server for OS/390 Release 3 provides the support required to run a Java transaction under the control of an OS/390 Java Virtual Machine (JVM). CICSplex SM supports this function by:

- Two new fields, JVM and JVMClass, on the BAS PROGDEF view.
- Three new fields, Runtime, JVM Class, and JVM Debug, on the operations PROGRAMD view.
- A new operations view, PROGRAMJ, a detailed view of the JVM Class value for the current program.

Open Transaction Environment

CICS Transaction Server for OS/390 Release 3 introduces an enhancement to the internal architecture of CICS to enable specified tasks to run under their own task control block. In support of this function, CICSplex SM extends its inquiry function for the following resource types:

- New fields, Force QR and Max open TCBs, added to the CICS RGN2 view.
- A new field, Concurrency, added to the PROGRAMD view.
- Amendments to the PROGRAM view.
- Amendments to the EXITGLUE and EXITTRUE views.
- Amendments to the TASK and TASKD views.

FEPI

CICS Transaction Server for OS/390 Release 3 introduces changes to the way in which the CICS/ESA Front End Programming Interface (FEPI) is managed using CICSplex SM. The changes are:

- New BAS views:
 - FENODDEF, which defines FEPI nodes
 - FEPOODEF, which defines FEPI pools
 - FEPRODEF, which defines FEPI property sets
 - FETRGDEF, which defines FEPI targets.
- FEPI resources are now installed using these new BAS views; they cannot be installed using the operations views

Support for Tivoli Global Enterprise Manager

CICS Transaction Server for OS/390 Release 3 provides support for Tivoli® Global Enterprise Manager (Tivoli GEM). For users of the Tivoli GEM client, the CICSplex SM Instrumentation feature enables users to view status information for the CMAS environment and the MAS environment of the CICSplex.

Changes made to this book for CICS Transaction Server for OS/390 Release 3

In addition to the changes required by new and changed function, the following changes have been made to this book:

- Information on CICSplex SM functions Business Application Services, workload management, real-time analysis, and monitoring has been removed from Chapter 2, and incorporated in the main chapters on these subjects.

Chapter 1. Introducing CICSplex SM

The CICSplex System Manager element of CICS Transaction Server for z/OS, Version 3 Release 1 (CICSplex SM) is a system management tool that enables you to manage multiple CICS systems from a single control point. Enterprises in which CICSplex SM may be needed range from those running 10 or 15 CICS systems, to those running two or three hundred (or more) CICS systems: in the latest MVS sysplex environment, having such large numbers of CICS systems to support a transaction-processing workload is becoming increasingly common.

CICSplex SM has the following key aspects:

- *CICSplex SM conforms to the IBM strategy for system management:*
 - It provides consistent creation and management of system management data.
 - It is an enterprise-wide system management application.
 - It supports increased automation of system management tasks.
 - It is structured into manager and agent components, has an enterprise information base, a coordination information base, and is based on a data model.
- *CICSplex SM has many reliability, availability, and serviceability (RAS) features.* For example:
 - User application CICS systems are managed by a discrete controlling system that has no associated terminals or use application elements or components. There can be multiple instances of this “managing” system.
 - User access to CICSplex SM functions is provided via a separate address space, of which there can also be multiple instances in an enterprise.
 - System management data is held separately in data-space caches, so that it is unaffected by what happens in both managing and managed CICS systems.
- *CICSplex SM is secure.*

Even though CICSplex SM has the potential to allow access to all of an enterprise’s CICS resources from a single session, it ensures that security is not compromised. You can use a SAF-compliant external security manager to prevent unauthorized access to:

- CICSplex SM’s own procedures and data
- CICSplex SM’s system management functions and the CICS resources they manage

Security checks that are usually performed by CICS (that is, CICS command checking and CICS resource checking) are performed instead by CICSplex SM.

Security setup is described for the IBM Resource Access Control Facility (RACF[®]), and is modeled on the use of RACF security in CICS. As with CICS, you can substitute another SAF-compliant external security manager if required.

Chapter 2, “The CICSplex SM environment,” on page 7 provides an overview of the basic CICSplex SM features and functions.

CICS system connectivity

This release of CICSplex SM can be used to control CICS systems that are directly connected to it.

For this release of CICSplex SM, the connectable CICS systems are:

- CICS Transaction Server for z/OS 3.1
- CICS Transaction Server for z/OS 2.3
- CICS Transaction Server for z/OS 2.2
- CICS Transaction Server for OS/390 1.3

You can use this release of CICSplex SM to control systems running supported releases of CICS that are connected to, and managed by, your previous release of CICSplex SM. However, if you have any directly-connectable release levels of CICS, as listed above, that are connected to a previous release of CICSplex SM, you are strongly recommended to migrate them to the current release of CICSplex SM, to take full advantage of the enhanced management services. See the *CICS Transaction Server for z/OS Migration from CICS TS Version 2.3* for information on how to do this.

Table 1 shows which supported CICS systems can be directly connected to which releases of CICSplex SM.

Table 1. Directly-connectable CICS systems by CICSplex SM release

| CICS system | CICSplex SM component of CICS TS 3.1 | CICSplex SM component of CICS TS 2.3 | CICSplex SM component of CICS TS 2.2 | CICSplex SM component of CICS TS 1.3 |
|------------------|--------------------------------------|--------------------------------------|--------------------------------------|--------------------------------------|
| CICS TS 3.1 | Yes | No | No | No |
| CICS TS 2.3 | Yes | Yes | No | No |
| CICS TS 2.2 | Yes | Yes | Yes | No |
| CICS TS 1.3 | Yes | Yes | Yes | Yes |
| TXSeries 4.3.0.4 | No | Yes | Yes | No |
| TXSeries 5.0 | No | Yes | Yes | No |

Features of CICSplex SM

This section describes briefly the system management functions of CICSplex SM:

- **A real-time, single-system image (SSI)**

CICSplex SM provides a real-time, single-system image (SSI) of all CICS regions and resources that make up your enterprise's transaction processing environment. CICSplex SM creates an inventory of the CICS systems and resources, and maps the relationships between them; this is called the CICSplex SM *topology*. Thus the operator no longer needs to know the location of a resource before working with it. The SSI is provided by the CICSplex SM *operating* function and applies to all CICSplex SM applications: operations, monitoring, real-time analysis (RTA), and workload management (WLM).

- **A single point of control**

The CICSplex SM operator can manage all of the enterprise CICS systems and their resources from a single session. That is, CICSplex SM can provide a *single point of control* for the enterprise, which in turn means that the CICSplex SM operator is able to manage large and complex configurations of CICS resources. Furthermore, you can have *multiple* points of control, so that multiple CICS

operators in multiple locations each may have a complete view of the enterprise CICS systems and resources, or a view of those systems and resources that is tailored to their individual requirements. Finally, because the physical location of these points of control is irrelevant, you have total flexibility in appointing control locations.

- **Management of your business applications**

CICSplex SM enables you to manage your enterprise in terms of your business applications rather than your CICS systems:

- **Business Application Services (BAS)** allows you to manage your resources at the application level, by providing:

- *A single point of definition* for your resources. All the resources for the CICSplex, and the relationships between them, are held in one place, on the CICSplex SM data repository, giving a single-system image for the CICSplex. CICSplex SM produces local and remote instances of a resource from a single definition.
- *Logical scoping*, which enables you to link and manage your resources according to their business relationship, rather than by their physical location in a CICS system.
- *Installation* of those resources, either automatically when the CICS system is started, or dynamically, as required, into a running CICSplex

BAS provides an alternative to resource definition online (RDO). RDO is based on a structure in which definitions are tied to a single group, and groups are processed sequentially from a group list. BAS frees resource definitions from associations with only one group. This means that a resource definition can be used and reused, and associated with more than one group as needed. BAS enables you to associate your resources according to their use within your enterprise. This in turn means that you can, if you wish, manage individual resources, rather than a group. For example, you could disable all the files associated with your payroll system, without affecting any other files in the CICSplex.

Chapter 5, “Managing resources using Business Application Services (BAS),” on page 41 gives an overview of BAS. For full details, see *CICSplex System Manager Managing Business Applications*.

- **Operations for the entire CICSplex**

From any point of control, the operator can take advantage of the SSI to perform tasks, either across the entire CICSplex or across a selected subset. You can display information about the current status of one or more instances of a CICS resource in an enterprise, and you can also change the status of the resources with a single entry. You can issue an *action command* to affect the status of the displayed resources.

You use the Web User Interface or the ISPF EUI to issue the commands. The responses are displayed in panels known as *operations views*, that can summarize related facts and provide access, using *hyperlinks* to other, related information. The operations views mirror the functionality currently provided for CICS systems. Operators can therefore work in essentially the same way as they are used to, without any change to their basic approach to daily system activities.

See the *CICSplex System Manager Operations Views Reference* *CICSplex System Manager Web User Interface Guide* for more information about operations views. ISPF EUI operations views are described in detail in *CICSplex System Manager Operations Views Reference*.

- **Management of your workloads**

CICSplex SM's workload management (WLM) uses the dynamic routing program EYU9XLOP to route eligible work requests from a requesting region to a suitable target region selected at the time the transaction is initiated.

EYU9XLOP handles the dynamic routing of:

- Transactions initiated at a terminal
- Eligible EXEC CICS START requests that are associated with a terminal
- Eligible EXEC CICS START requests that are not associated with a terminal
- Dynamic program link (DPL) requests that are received using:
 - The CICS Web Interface
 - The CICS Gateway for Java
 - External CICS interface (EXCI) client programs
 - Any CICS client workstation products using the External Call Interface (ECI)
 - Distributed Computing Environment (DCE) remote procedure calls (RPCs)
 - Open Network Computing (ONC) RPCs
 - Internet Inter-Object Request Block Protocol (IIOP)
 - Any function that issues an EXEC CICS LINK PROGRAM request
- Transaction associated with CICS business transaction services (BTS) activities

CICSplex SM provides management support for BTS by dynamically routing BTS work across a BTS-set, and by monitoring the management of data in a distributed environment. For full details, see *CICS Business Transaction Services*.

You do not have to use CICSplex SM workload management to route BTS activities, but using CICSplex SM can offer many benefits:

- Management of distributed data
 - Workload separation and balancing functions of workload management
 - Dynamic routing program EYU9XLOP
 - Reduction in link definitions if you also use BAS
 - Cooperation between BAS and BTS in the management of your business environment
- Enterprise beans executing in CICS-provided CorbaServers

You can write your own program, and use it in conjunction with the CICS-supplied user-replaceable programs DFHDYP and DFHDSRP, to handle your dynamic routing requirements.

More information on dynamic routing and workload management is given in Chapter 6, "Managing workloads," on page 57. You could also refer to the relevant CICS Transaction Server for z/OS, Version 3 Release 1 publications.

- **Automated exception reporting for CICS resources**

CICSplex SM's Real-Time Analysis (RTA) function provides automatic, external, notification of conditions in which you have expressed an interest. The notification can take the form of a console message, or of a generic alert to NetView®, or both. Real-time analysis is not concerned solely with commonly recognized error conditions: you can ask to be notified about any aspect of a resource's status. Real-time analysis also enables you to take actions without having to use an external automation product.

The RTA functions of CICSplex SM are described in greater detail in Chapter 7, "Exception reporting using real-time analysis (RTA)," on page 73. For full details, see *CICSplex System Manager Managing Resource Usage*.

- **Monitoring functions for the collection of statistical data for CICS resources**

The CICSplex SM *monitoring* functions support the collection of performance-related data, at user-defined intervals, for named resource instances within a set of CICS systems.

The monitoring functions of CICSplex SM are described in more detail in Chapter 8, “Collecting statistics using CICSplex SM monitoring,” on page 83. For full details, see *CICSplex System Manager Managing Resource Usage*.

- **An application programming interface (API)**

CICSplex SM provides an application programming interface (API) that allows applications to:

- Access information about CICS and CICSplex SM resources. For details of system connectivity, see “CICS system connectivity” on page 2.
- Invoke the services of CICSplex SM.

A command-level interface is available to programs that are written in these languages:

- Assembler
- PL/I
- COBOL
- C

Programs written in Assembler, PL/I, COBOL, and C are supported in the CICS/ESA, CICS/MVS, and MVS/ESA Batch, TSO, and NetView environments.

In addition, a REXX run-time interface is available under MVS/ESA Batch, TSO, and NetView.

You can use the CICSplex SM API to write external programs that automate the management of CICSplex SM and CICS resource definitions. Such programs could be used to integrate the CICSplex SM system management functions into your enterprise-wide change management process. For example, you could write an API program to coordinate resource definition changes with database or file updates, or the standard life cycle of an application. For a complete description of the API, see the *CICSplex System Manager Application Programming Reference* and the *CICSplex System Manager Application Programming Guide*.

- **Management of the CICSplex SM environment**

You manage the CICSplex SM environment using:

- **CICSplex SM objects**

To define the configuration of your CICS systems to CICSplex SM (and to define your BAS, WLM, RTA, and monitoring requirements) you create *CICSplex SM objects*, and associate them with each other. For each object, and for each association or link between them, a record is created in a CICSplex SM data repository. CICSplex SM objects are described in “CICSplex SM objects” on page 13.

- **Data repository**

The data repository contains the objects that define the CICSplex SM components, resources, system management requirements, and the relationships between them. The definitions can be created using the Web User Interface or EUI views, the CICSplex SM API, or the *batched repository-update facility*.

- **The batched repository-update facility**

The batched repository-update facility allows you to create and update large numbers of CICSplex SM and CICS resource definitions by submitting one command that is used as the template for other definitions. The batched repository-update facility is also used for migrating your definitions from one platform to another, and for backing up the data repository. More information

is given in “The batched repository-update facility” on page 15. For details, see *CICSplex System Manager Administration*.

- **Management of time-dependent activity**

Much of CICSplex SM’s activity is time dependent. For example, you can specify on your RTA and monitoring definitions when you want the definition to be active. Also, you may want CICS systems running in the same CICSplex but different time zones to run as if they were in the same time zone. You are able to create time period-definitions that control:

- Exactly when any part of your enterprise is operational, regardless of the local time zone.
- The times you want certain system management functions to be operational.

The international standard for time zones is used, based on Greenwich Mean Time (GMT). You select the time zone in which you want your CICSplex to run. You can then make adjustments, either for locations that have implemented times that are not different from GMT by 60-minute multiples, and for daylight saving.

Details of time-period definition are in *CICSplex System Manager Administration*.

- **Integration with Tivoli Business Systems Manager**

For users of Tivoli Business Systems Manager, the CICSplex SM Instrumentation feature supplied with that product enables users to view status information for the CMAS and MAS environments of the CICSplex. This can be very helpful in reporting back the status of specific subsets of an enterprise’s applications. For more information see “Support for Tivoli Business Systems Manager” on page 15.

Where next?

Chapter 2, “The CICSplex SM environment,” on page 7 describes the configuration and topology of CICSplex SM and the system management functions in more detail. Chapter 3, “Designing your CICSplex SM environment,” on page 17 starts guiding you through the process of designing your own CICSplex SM environment.

A Starter Set is supplied that contains all you need to get started. See Chapter 10, “The CICSplex SM Starter Set,” on page 101 for the details, including the CICSplex SM and CICS resource definitions. You might find it useful to refer to these for examples as you read the following chapters.

Chapter 2. The CICSplex SM environment

This chapter is intended for anyone who needs an overview of CICSplex SM. This chapter looks at the structure of CICSplex SM and at the components that enable CICSplex SM to perform its tasks. This chapter introduces all of the components of CICSplex SM that you need to know about, either because you will have to define them at some stage, or because their very presence or mode of operation will affect the design of your CICSplex SM environment.

Figure 1 on page 8 shows the main components of CICSplex SM and illustrates some of the links between them. The figure is explained in the descriptions of the CICSplex SM components in the remainder of this chapter.

This chapter covers:

- “Overview”
- “The CICSplex” on page 9
- “The managed application system (MAS)” on page 9
- “CICSplex SM address space (CMAS)” on page 10
- “Environment Services System Services (ESSS)” on page 11
- “The coordinating address space (CAS)” on page 12
- “The CICSplex SM end user interface (EUI)” on page 12
- “The CICSplex SM Web User Interface” on page 11
- “CICSplex SM objects” on page 13
- “The data repository” on page 15
- “The batched repository-update facility” on page 15
- “Support for Tivoli Business Systems Manager” on page 15
- “Where next?” on page 16

Overview

This section gives you a brief overview of the main elements of the CICSplex SM environment. See Figure 1 on page 8.

Each CICS system to be managed by CICSplex SM is called a *managed application system (MAS)*. The MASs are defined and managed as part of a *CICSplex*. Each MAS in a CICSplex is managed by a *CICSplex SM address space (CMAS)*. For the CICSplex SM component of CICS Transaction Server for z/OS, Version 3 Release 1, this must be a CICS Transaction Server for z/OS, Version 3 Release 1 system running on an MVS image (the release level of CICSplex SM and the CICS system of the CMAS must be the same). For more information about system connectability, see “CICS system connectivity” on page 2.

The MASs in a CICSplex can be managed by several CMASs, but only one is defined as the *maintenance point CMAS* responsible for keeping the data used by each CMAS synchronized.

CMASs communicate across *CMAS-to-CMAS* links, which are typically used for routing management commands and data between CMASs. A *coordinating address space (CAS)* is used to set up the CICSplex SM component topology, and to support the MVS/TSO ISPF end-user interface (EUI) to CICSplex SM.

The CICSplex SM Web User Interface is an alternative to the MVS/TSO ISPF end-user interface. The Web User Interface server runs on a dedicated CICSplex SM local MAS at the same release level connected to a CMAS. See “The CICSplex SM Web User Interface” on page 11 for more information.

Resource definition is managed through Business Application Services (BAS). Workload management (WLM), real-time analysis (RTA), and monitoring services are used to manage the CICSplex SM configuration and gather statistical information.

All CICSplex SM components, resources, system management requirements, and the relationships between them, are held as *objects* (see “CICSplex SM objects” on page 13) in a *data repository* (see “The data repository” on page 15). These objects can be manipulated using the end-user interface views (see “The CICSplex SM end user interface (EUI)” on page 12). The batched repository-update facility (see “The batched repository-update facility” on page 15) is provided for the batched creation of CICSplex SM resource definitions.

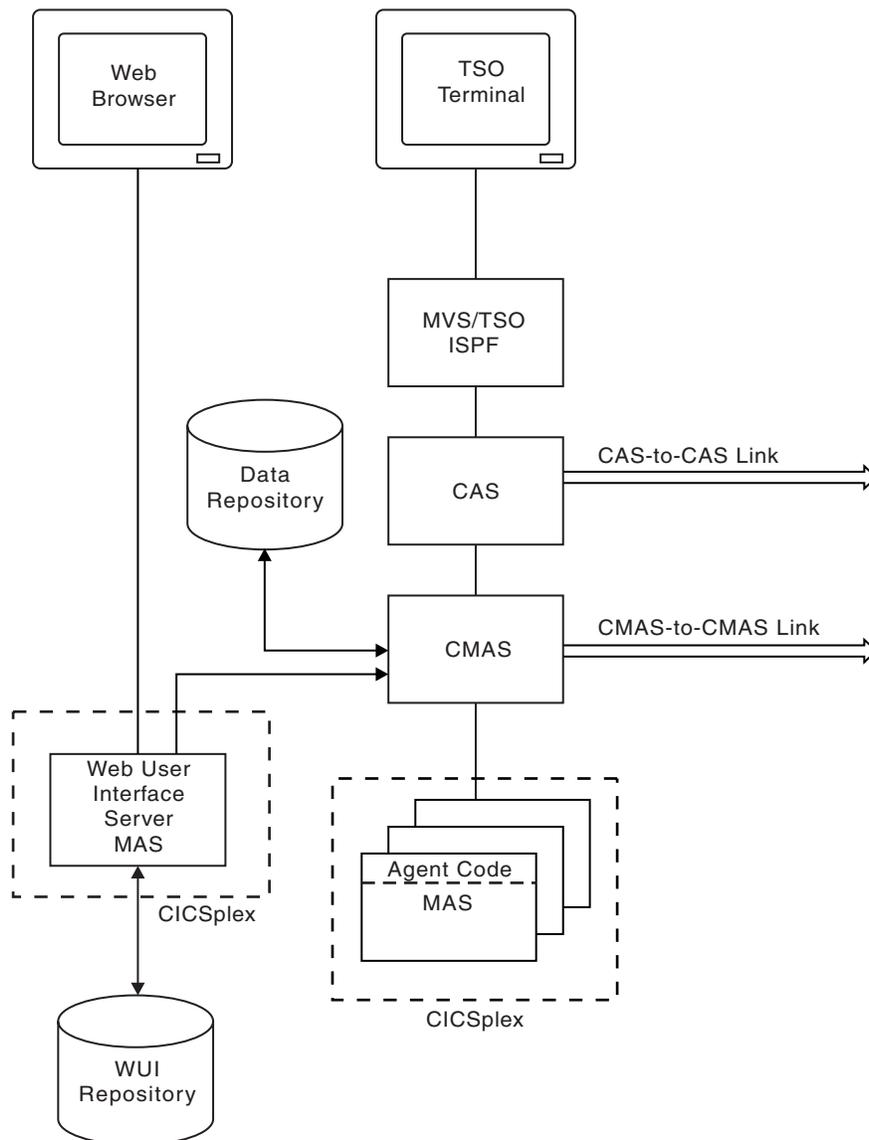


Figure 1. Key components of a CICSplex SM configuration. Each of these components is described in this chapter.

The CICSplex

For CICSplex SM's purposes, a *CICSplex* is **any** grouping of CICS systems that you want to manage and manipulate as if they were a single entity. That is, a CICSplex is a management domain, made up of those CICS systems for which you want to establish a single system image (SSI). A CICSplex managed by CICSplex SM could include every CICS system in your enterprise. Alternatively, you could define multiple CICSplexes, each of which would include a logical grouping of CICS systems. For example, a CICSplex could comprise all CICS systems on a particular MVS image, or all CICS systems accessible by a subset of your users, or all CICS systems serving a particular geographical area. Furthermore, the composition of a CICSplex can be altered without affecting the functions of the underlying CICS systems. The CICS systems in a single CICSplex managed by CICSplex SM do not have to be explicitly connected to each other for management purposes.

The most significant facts about a CICSplex managed by CICSplex SM are:

- The CICSplex is the largest unit you can work with. That is, you cannot group CICSplexes and manipulate such a group as a single entity.
- You cannot copy CICSplex SM data from one CICSplex to another. For system management purposes, the CICSplex is “sealed” against other CICSplexes.
- CICSplexes are mutually exclusive, so no CICS system can belong to more than one CICSplex.

CICSplex SM enables you to define subsets of a CICSplex, which are known as *CICS system groups*. CICS system groups are not mutually exclusive, and can reference the same CICSplex SM definitions. Thus, if you decide to include every CICS system in your enterprise in a single CICSplex, there are mechanisms for managing groups of CICS systems within the CICSplex as if each group were a single system.

You can assign an unlimited number of CICS systems and CICS system groups to an existing CICSplex.

Although you can define a CICS system to only one CICSplex, you can assign a CICS system to multiple CICS system groups within the CICSplex. You can also assign the CICS system group to any number of other CICS system groups.

The managed application system (MAS)

Each running CICS system that is being managed by CICSplex SM is known as a *managed application system (MAS)*. CICSplex SM can manage the systems listed in “CICS system connectivity” on page 2.

If a MAS belongs to a CICS/XRF pair, it must be the active region.

All the MASs within a CICSplex are managed by the same *CICSplex SM Address Space (CMAS)* or the same group of CMASs.

the managed application system (MAS)

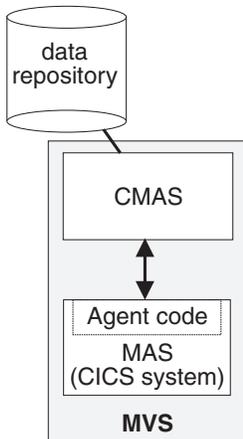


Figure 2. MASs and their CMAS. Each of these components is described in this section.

Each MAS contains CICSplex SM *agent code* that implements CICSplex SM function, such as data collection, for the CMAS by which it is managed. For example, if resource monitoring is active for a particular MAS, agent code in the MAS monitors the selected resources and feeds the resulting data to the CMAS.

The MASs within a single CICSplex do not have to be explicitly connected to each other for CICSplex SM-specific communication. However, CICS connections required in support of transaction routing and function shipping are still required.

MASs can be defined and managed as individual CICS systems or grouped into *system groups* within the CICSplex. Each system group is one or more CICS systems that you want to manage as a unit subset of the CICSplex. System groups are described in “Identifying system groups” on page 21.

In CICS Transaction Server for z/OS, Version 3 Release 1 all MASs are local, that is they run on the same MVS image as the CMAS by which they are managed. You do not need to define an explicit link between the CMAS and the local MAS. For local MASs, system-management data is accumulated in data-space caches and is communicated to the CMAS via the Environment Services System Services (ESSS) address space (which is described on page “Environment Services System Services (ESSS)” on page 11).

| Remote MASs running on a different MVS image from the managing CMAS or on
| other operating systems such as Windows or OS/2 are not supported in CICS
| Transaction Server for z/OS, Version 3 Release 1 or later.

A MAS can be set up as a CICSplex SM Web User Interface server. In this case, the CICS release level of the MAS and the CICSplex SM Web User Interface server must be the same.

CICSplex SM address space (CMAS)

The *CICSplex SM address space (CMAS)* is the hub of any CICSplex SM configuration, because it is responsible for most of the work involved in managing and reporting on CICS systems and their resources. Every CICSplex is managed by at least one CMAS. The CMAS is responsible for the *single system image (SSI)* that enables the operator to manage a CICSplex as if it were a single CICS system, regardless of the number of CICS systems defined as belonging to the CICSplex, and regardless of their physical location.

CICSplex SM address space (CMAS)

The CMAS implements the BAS, WLM, RTA, and monitoring functions of CICSplex SM, and maintains configuration information about the CICSplexes it is managing. It also holds information about its own links with other CMASs. It stores this information in its *data repository*. (See Figure 2 on page 10).

A CMAS is a full-function CICS Transaction Server for z/OS, Version 3 Release 1 system. Most CMAS components run as CICS tasks, and CMAS connections to other components are implemented using CICS intercommunication methods.

Note: The CMAS does not support user applications or terminals, and its resources should not be considered available for non-CMAS tasks, including (without limitation) the use of any monitoring and performance tool other than those supplied as a part of CICSplex SM.

A CMAS cannot be running a lower release of CICS than its MASs. Also, both the CMAS and the MASs must be running the same release of CICSplex SM.

A CMAS is not part of any CICSplex: a single CMAS can participate in the management of multiple CICSplexes, but belongs to none of them.

If a CICSplex is managed by multiple CMASs:

- The CMASs are connected to each other by *CMAS-to-CMAS* links defined to CICSplex SM. These links ensure that each CMAS can access data held by other CMASs, and a single-system image can be presented to the operators.
- One of the CMASs is designated the *maintenance point* CMAS. That CMAS is responsible for maintaining all the CICSplex SM definitions relating to a CICSplex and keeping all the data repositories in synchronization.

CICSplex SM can issue SNA generic alerts to NetView, provided that a CMAS is installed on the same MVS image as the NetView instance with which CICSplex SM is working.

Environment Services System Services (ESSS)

Environment Services System Services (ESSS) is a limited-function, MVS/ESA system address space that provides MVS services to CICSplex SM components. In particular, ESSS owns all of the MVS/ESA data spaces on an MVS image, so that they can exist independently of CMASs and MASs, yet remain accessible by both. The benefit of this arrangement is that the CICSplex SM data accumulating in the data spaces is not vulnerable to events in the MAS and CMAS components. ESSS also plays a part in some aspects of communication between a CMAS and any NetView instance on the same MVS image as the CMAS.

There is one instance of ESSS for each version or level of CMAS on any MVS
image.

The CICSplex SM Web User Interface

| The CICSplex SM Web User Interface (WUI) is a customizable,
| platform-independent interface for your web browser, developed to replace the
| CICSplex SM ISPF End User Interface (EUI). It was originally designed primarily for
| use in operations and help desk environments but in CICS Transaction Server for
| z/OS, Version 3 Release 1 has been enhanced to match the functionality of the end
| user interface as well as being easier to use. You can link to the Web User
| Interface from any location that can launch a web browser.

the CICSplex SM Web User Interface

The server component of the Web User Interface runs on a dedicated CICS Transaction Server for z/OS, Version 3 Release 1 CICSplex SM local MAS connected to a CMAS at the same release level.

The Web User Interface is supplied with a set of linked menus and views but can also be customized to reflect your business procedures and to suit the needs of individual users.

Information about your enterprise is displayed in formatted displays called *views*. You navigate between views control the type of information that is displayed using hyperlinks on views and menus. From a view, you use *action buttons* to manage the displayed data.

The general principles of the WUI are described in *CICSplex System Manager Web User Interface Guide*.

The coordinating address space (CAS)

The *coordinating address space* (CAS) is an MVS subsystem whose main function is to support the MVS/TSO ISPF end-user interface (EUI) to CICSplex SM. The CAS is not part of a CICSplex, but belongs to the managing topology of CICSplex SM. (See Figure 2 on page 10).

A CAS need not be started on every MVS system where a CMAS runs to provide Single System Image (SSI) function for EUI requests directed at a CICSplex. However in order for an EUI user to set the context to a CMAS, that CMAS must be connected to an active CAS.

The CICSplex SM end user interface (EUI)

The CICSplex SM user interface is an MVS/TSO ISPF EUI supported by the CAS. This ISPF interface is used to set up the CICSplex SM component topology and can be used by operators to manage CICS systems and resources. CICSplex SM's EUI is used:

- By the CICSplex SM operator, who uses it much as the master-terminal operator uses CEMT and other, CICS-supplied transactions to monitor and control the resources of a single CICS system.
- By the system administrator, who uses the interface to define and maintain the CICSplex SM configuration, and to specify BAS, WLM, RTA, and monitoring requirements.

The general principles of the CICSplex SM user interface are defined in the *CICSplex System Manager User Interface Guide*.

Information about your enterprise is displayed in formatted displays called *views*. You control the type of information that is displayed, and the format in which it is displayed, using *view commands*. From a view, you use *action commands* to manage the displayed data. Before you enter a view command, you define the CICS systems or resources affected by the command by setting a *context* and a *scope*:

- The *context* is almost always the CICSplex.

Note: When you are viewing or updating CMAS configuration data, the context is a CMAS rather than a CICSplex, and any scope value is ignored. (Remember that the CICSplex is the largest single unit on which any CICSplex SM command can act.)

- The *scope* is always a subset of the context: its purpose is to limit the effects of CICSplex SM commands to particular CICS systems or resources. Indeed, the effects of any command cannot be felt beyond the current scope. The scope can be a single CICS system, a CICS system group (made up of any number of CICS systems in the CICSplex), or a business application (represented by a *resource description*).

You can look at CICS systems or resources anywhere in your enterprise from one point, the single point of control, simply by changing the context and scope.

CICSplex SM objects

To define the configuration of your CICS systems to CICSplex SM, and to define your BAS, WLM, RTA, and monitoring requirements, you create *CICSplex SM objects*, and associate them with each other. For each object, and for each association or link between them, a record is created in a CICSplex SM data repository. Figure 3 on page 14 shows how the CICSplex SM objects relate to each other.

CICSplex SM objects

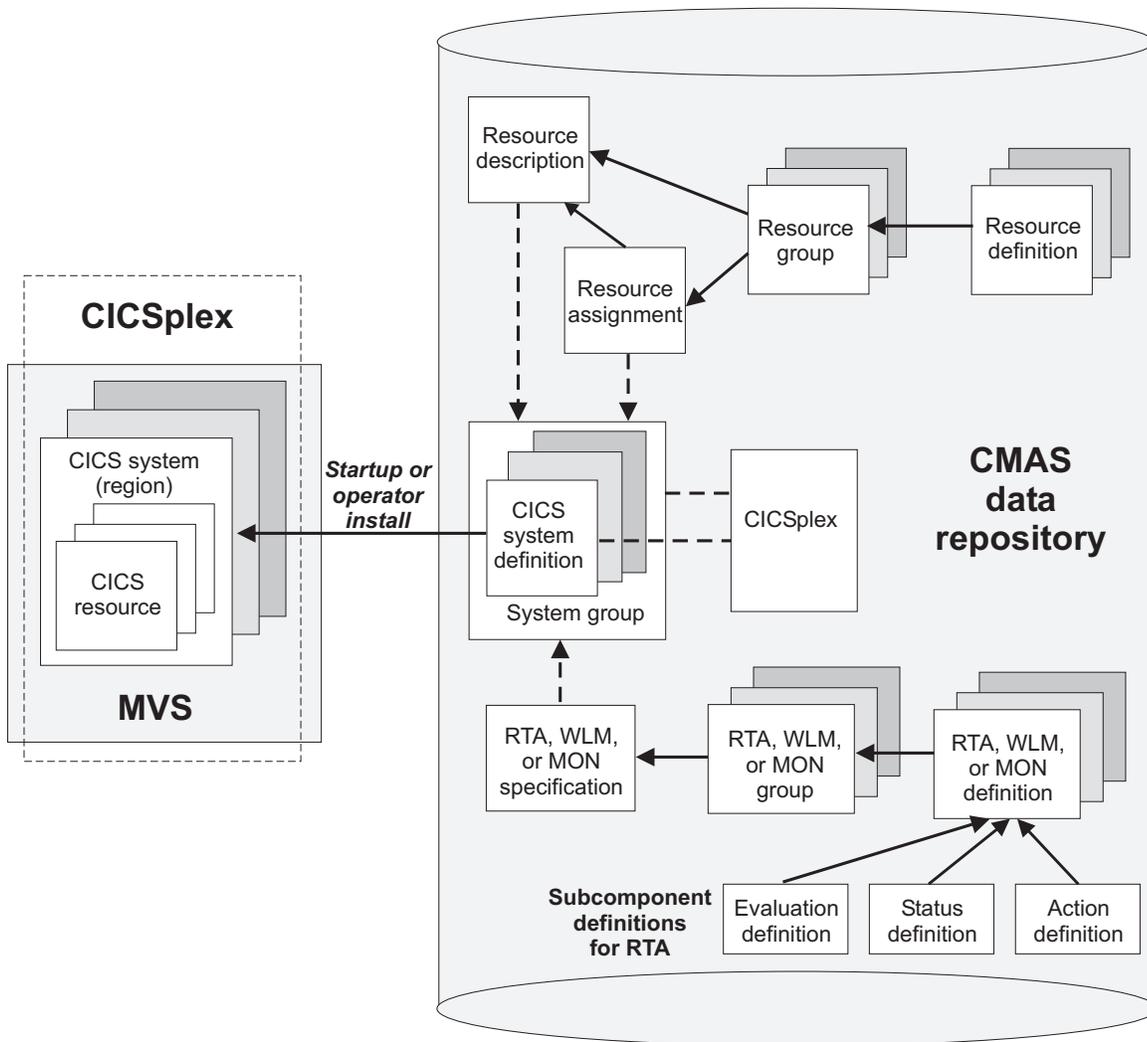


Figure 3. The CICSplex SM object model

These objects can be split into three categories:

1. *CICSplex and CICS system* objects, used for defining the CICSplexes and CICS systems to be managed by CICSplex SM. This includes defining the links between CICS systems and creating CICS system groups. These objects are described in Chapter 3, “Designing your CICSplex SM environment,” on page 17; information on defining these objects is in Chapter 4, “Defining the CICSplex SM configuration and topology,” on page 35.
2. *BAS* objects, used for logical scoping and managing the CICS resource definition and installation processes. You can look on the objects in this category as defining what resources you want your CICS systems to use. These CICSplex SM objects are described in “Multiple versions of a resource definition” on page 42. Information on defining BAS objects is in “Using full-function form BAS” on page 50.
3. *Operation* objects are those objects used for operating the CICS resources that exist in running CICS systems. You can look on the objects in this category as identifying the resources you want to monitor at run-time, for automated workload management, automated exception reporting, and collection of

statistical data. These CICSplex SM objects are described in “Managing monitoring functions” on page 84, “Managing workloads” on page 65, and “Managing RTA” on page 77.

The data repository

The single-system image of a CICS system or group of CICS systems is provided by the CMAS using the CICSplex SM and resource definitions held on the *data repository*. Each CMAS has its own data repository. The data repository is a VSAM KSDS that is created using a CICSplex SM post-installation job. As with all data sets, you need to take regular backups of each data repository in your environment. Details of creating and managing a data repository are in the *CICS Transaction Server for z/OS Installation Guide*.

CICSplex SM definitions and CICS resource definitions held on the data repository can be managed in the following ways:

- Using the EUI views; see “The CICSplex SM end user interface (EUI)” on page 12
- Using the batched repository-update facility; see “The batched repository-update facility”
- Using the API; see the *CICSplex System Manager Application Programming Reference* and *CICSplex System Manager Application Programming Guide*.

You can create a visual map of the definitions in your data repository. The map can be of business application services, resource monitoring, real-time analysis, or workload monitoring. You select a starting point for the map, which might be, for example, a CICS system group or an individual resource, and CICSplex SM displays the starting point and all the definitions that either refer to that definition or are referred to by it. You use either the MAP view command or the batched repository-update facility to create a map.

The batched repository-update facility

To help you in the migration of your CICS definitions to CICSplex SM, you can use the EXTRACT command of the CICS DFHCSDUP utility to read CSD records. CICSplex SM provides an exit routine EYU9BCSD that generates, from the DFHCSDUP output, equivalent resource definitions for input to the batched repository-update facility. For more information, see *CICSplex System Manager Managing Business Applications*.

The batched repository-update facility is also useful for migrating your definitions from one platform to another. You can use it to retrieve existing resource definitions from the CICSplex SM data repository then, after making any required changes to the definitions, input the changes to another batched repository-update facility run that creates resource definitions on the new platform.

For a complete description of the batched repository-update facility, see *CICSplex System Manager Administration*.

Support for Tivoli Business Systems Manager

Tivoli Business Systems Manager provides a consistent systems management view of an enterprise. Most enterprises have a set of business objectives. Those business objectives can be achieved using various business systems. A business system might comprise various hardware and software packages from multiple vendors, as well as user-written applications. One or more of the applications within

support for Tivoli Business Systems Manager

a business system can make use of various middleware components such as database, messaging services, or a transaction monitor.

If your business system uses the CICS transaction monitor managed by CICSplex SM, Tivoli Business Systems Manager provides an agent to integrate with CICSplex SM systems management.

The agent provided by Tivoli Business Systems Manager enables an individual using the Tivoli Business Systems Manager client to view status information for CICSplex SM-managed resources. These resources include the CICSplex, CMAS, MAS, transactions and files.

Tivoli Business Systems Manager uses the CICSplex SM API, EXEC CPSM, to discover the CICSplex SM objects it monitors. Tivoli Business Systems Manager collects events generated as a result of SAM (Systems Automation Monitoring) and RTA (Real Time Analysis) processing, in order to monitor the status of these resources.

Tivoli Business Systems Manager also provides the user with the ability to link to the CICSplex SM Web User Interface.

For details of the Tivoli Business Systems Manager agent for CICSplex SM, see the *Tivoli Business Systems Manager CICSplex SM Release Notes*.

Where next?

Chapter 1, “Introducing CICSplex SM,” on page 1 and this chapter have given you an overview of CICSplex SM and the facilities it offers to help you administer your enterprise. The rest of this book tells you how to design your CICSplex SM environment and how to start planning for and defining those functions you wish to use. The chapters point you to where you can find the details of how to perform the required tasks.

Chapter 3. Designing your CICSplex SM environment

This chapter takes you through the process of identifying the CICSplexes in your enterprise, the CICS systems that will be managed by CICSplex SM (the MASs) and those that will not.

The chapter covers:

- “Designing your CICSplexes”
- “Locating CMASs” on page 24
- “Locating CASs” on page 30
- “Naming your CICSplex SM entities” on page 33
- “A staged implementation” on page 34
- “Updating the initial design” on page 34
- “Where next?” on page 34

This chapter helps you to design your CICSplex SM environment by giving you guidance on each step of the process. An example is used to illustrate the process.

Designing your CICSplexes

The CICSplex is the largest unit that you can manipulate in your CICSplex SM configuration. A CICSplex is made up of an association of CICS systems and CICS system groups. This section gives guidance on deciding how to group your CICS systems into system groups, and then to identify CICSplexes.

Identifying your CICS systems and regions

The very first thing you must do when planning to install CICSplex SM is to identify the CICS systems or regions in your enterprise. You may already have a clear picture of the systems you have, and of where they are installed. However, in the larger enterprises, where CICS systems are numbered in the hundreds, it's possible that no one individual has this complete view. Whatever the case, the aim of this exercise is to *document* the current arrangement of your CICS systems in a graphical form. The “map” you produce should be a logical representation of your CICS systems rather than a physical one, so don't worry unduly about recording where particular processors are located, for example. Whether you simply sketch the map on paper or use an on-line graphics tool, be sure to leave plenty of space so that you can update the map with CICSplex SM components as you work through this chapter.

Figure 4 on page 18 shows an example of the type of map you should be aiming to produce.

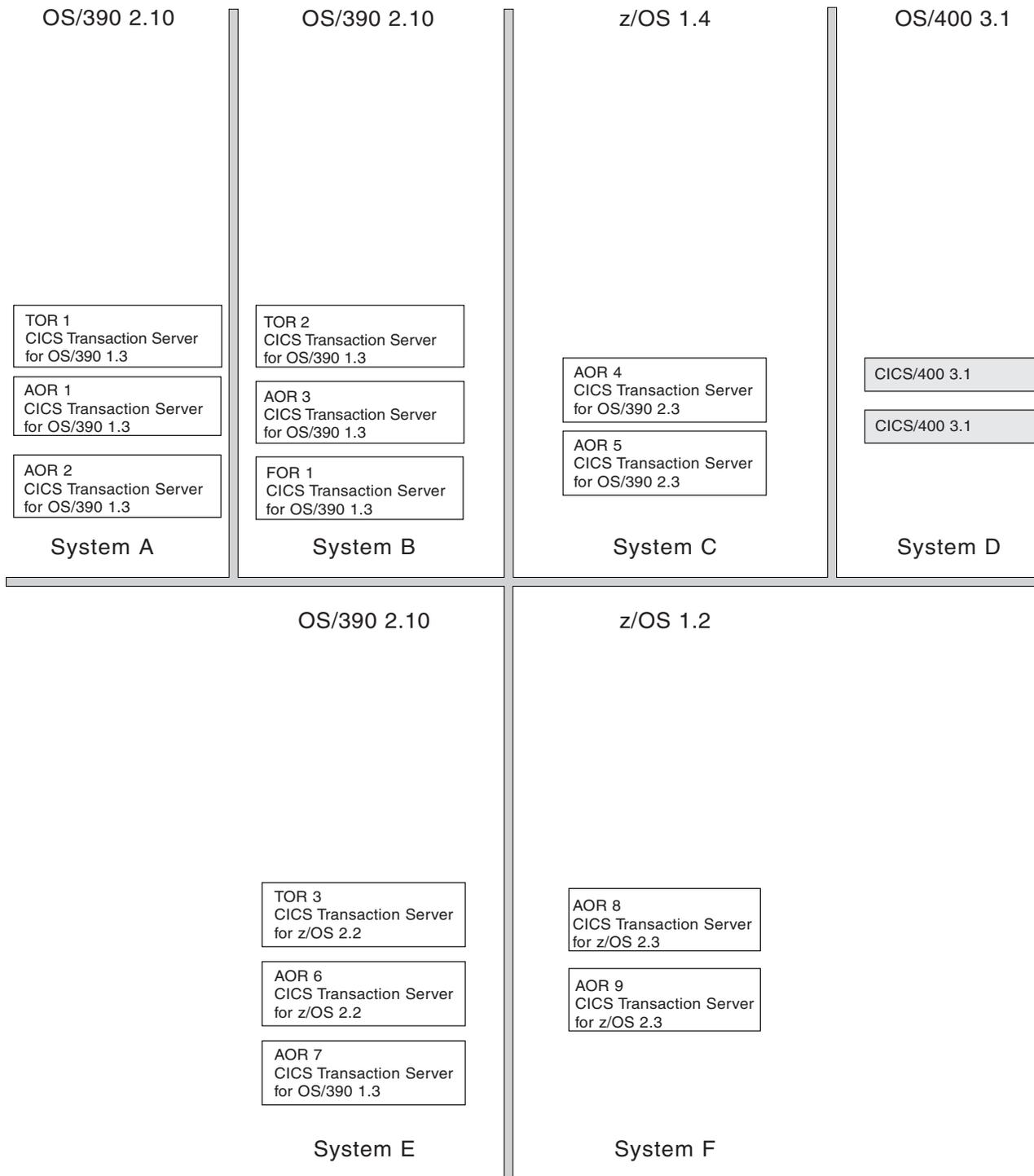


Figure 4. A map of the enterprise CICS systems. This enterprise has 15 CICS systems distributed among OS/390, z/OS, and OS/400, environments.

Your initial map of the enterprise CICS systems should include every operating environment in which CICS is installed. It should also show:

- The current version and release of the operating systems being used
- The CICS systems running in each environment, and the CICS platform, version, and release of each one
- The resource-manager role of each CICS system, if applicable

(If you can't get all of this detail on your map, record it separately from the graphical representation of the CICS systems.)

Which CICS systems can be managed by CICSplex SM?

Next, refine the map by identifying those CICS systems or systems that can and cannot be managed by CICSplex SM. A list of CICS systems that CICSplex SM can manage is given in "CICS system connectivity" on page 2. Any CICS systems that are not directly-connectable to CICS Transaction Server for z/OS, Version 3 Release 1 and later will need special consideration when you are locating CMASs (see "Locating CMASs" on page 24).

In the example map shown in Figure 4 on page 18, only the CICS systems running under OS/400® cannot be managed by CICSplex SM. On your own map, choose a convention, such as shading or use of color, for marking CICS systems that cannot be managed by CICSplex SM. However, don't remove them from the map altogether. If you decide to move those systems to a CICS platform or release that CICSplex SM can manage, they can be reinstated easily in the enterprise map.

All the other CICS systems become your MASs.

How many CICSplexes are there in your enterprise?

When you have identified those CICS systems or regions in your enterprise that can be managed by CICSplex SM, your next task is to decide how many CICSplexes you want to define to CICSplex SM, and which of your CICS systems is to belong to each. You can have any number of CICSplexes. For example, you could define:

- One CICSplex for the entire enterprise, or
- One CICSplex for each geographical location, or
- One CICSplex for each business unit within the enterprise, or
- One CICSplex for each existing TOR-AOR-FOR configuration, or
- One CICSplex for each processor, or
- One CICSplex for each operating environment (OS/390 and z/OS)

If you do not plan to use workload management facilities, there are no restrictions on how you combine CICS systems and CICS system groups to form a CICSplex. For example, you might associate CICS systems by:

- Geographic area within the CICSplex.
- Function, such as all CICS systems that are application-owning regions (AORs), file-owning regions (FORs), or terminal-owning regions (TORs).
- Application, such as the CICS systems serving as AORs, FORs, and TORs that are used by a specific application or group of applications.
- Time period, such as all CICS systems that are normally active during specific hours of the day or night.

If you do plan to use workload management facilities, you must ensure that:

- Each CICS system that is to act as a target for specific workload processing must be in the same CICSplex as the CICS systems acting as routing regions. (The routing regions and target regions associated with a CICSplex can be in the same or different MVS images.)
- CICS systems acting as the routing regions must be CICS TS systems.

If you plan to use the logical scoping, resource management, or installation functions of BAS, you should keep a business application within one CICSplex.

If you plan to use CICS BTS, you should keep a BTS-set with one CICSplex.

The question is, how do you decide what to do? There are no hard-and-fast rules governing the number of CICSplexes you define, but there are some guidelines that will help you select the most suitable configuration for your enterprise. These guidelines are presented in the form of a three-step process:

1. Decide to have a single CICSplex for the entire enterprise
2. Look for reasons why a single CICSplex might not be feasible
3. Confirm your decision

Each of these steps is discussed below.

Step 1: decide to have a single CICSplex for the entire enterprise

This may seem an arbitrary recommendation, but in fact the majority of enterprises will discover that having a single CICSplex best suits their system management goals.

You will remember that the CICSplex is the largest single entity that CICSplex SM can manipulate, and that none of the CICSplex SM definitions or specifications can cross a CICSplex boundary. Furthermore, CICSplexes are mutually exclusive, so no CICS system can belong to more than one CICSplex. Therefore, having a single CICSplex for the enterprise brings several advantages. For example:

- It provides the greatest opportunity for sharing and reusing BAS, WLM, RTA, and monitoring specifications and definitions.
- It allows you the most flexibility in managing the CICS workload when using the CICSplex SM workload balancing functions.
- Because the SSI operates at the CICSplex level, it gives an SSI of the *enterprise* CICS systems. Also, the CICS operator is able to have a complete view of the enterprise CICS resources from a single CICSplex SM window. (Because any CICSplex SM window can display data from only a single context—that is, a single CICSplex—multiple windows would have to be displayed in a multi-CICSplex environment.)

In summary, having one CICSplex means that there are no system management barriers between one group of the enterprise CICS systems and another.

Step 2: look for reasons why a single CICSplex might not be feasible

The single CICSplex solution is not going to be the best approach in every enterprise, either because its implementation simply isn't possible, or because it doesn't accord with other system management goals.

Firstly, you must ask whether the organization of your enterprise lends itself to a single CICSplex. For example, if you have processors in different geographical locations, are there connections between those processors, or are they managed as separate entities, each with its own workload? If you have these separate units in your enterprise, it's likely that you will need to define multiple CICSplexes, and so manage the enterprise CICS systems as if they belonged to more than one enterprise.

Similarly, is your enterprise organized and run as multiple, separate business units? For example, if you are running a bureau that provides computing services to a variety of customers, the absolute separation of one set of regions from another, even within a single processor, might simplify other processes, such as security management, customer billing, or workload management. If you have similar reasons for wanting to keep the

management of some regions entirely separate from the management of others, you should define multiple CICSplexes rather than one.

If you have decided that you need to define more than one CICSplex, for reasons such as those outlined above, it's probably obvious to you already which CICS system or CICS systems should belong to each. If it isn't, you should revisit your decision to have multiple CICSplexes because it suggests that you are trying to erect artificial barriers. And, as a final check, you should ensure that the way you separate the regions is not disruptive to your other system management goals. For example, if you want to use CICSplex SM's WLM functions, both routing regions and the target regions to which they route transactions must belong to the same CICSplex, unless you are planning to customize the supplied dynamic routing program.

Step 3: confirm your decision

When you have decided whether to have one CICSplex or many, check your decision against these other considerations:

- Your CICSplex organization should mirror your enterprise organization: if your enterprise is structured as multiple, independent units, having multiple CICSplexes is probably the better approach; if it's structured as a single entity, the single-CICSplex solution is likely to be the more suitable.
- Does your decision conflict with your enterprise's plans, either for its business or its information systems? For example, if your enterprise currently operates as multiple, separate entities, are there plans to unify them at any stage?
- Is your proposed configuration as simple as it could be? For example, are you planning to define four CICSplexes when two would support your system management goals?
- If you are planning multiple CICSplexes, have you considered whether CICS system groups could be used to achieve your goals? CICS system groups, unlike CICSplexes, are not mutually exclusive. This can be an advantage or a disadvantage, depending on your reasons for having multiple CICSplexes.

Finally, try to remember that you can alter your decision. Ideally, you would discover the best possible configuration at your first attempt. However, if you decide after a while that a different CICSplex configuration would be better, you can make the necessary changes.

Figure 5 on page 23 shows the number of CICSplexes required in the example enterprise.

Identifying system groups

This section gives some suggested CICS system groups for the example configuration:

CICSplex 1

- Group 1: TOR 1 and TOR 2
- Group 2: AOR 1 through AOR 5
- Group 3: FOR 1
- Group 4: All CICS systems on System A
- Group 5: All CICS systems on System B
- Group 6: All CICS systems on System C
- Group 7: Group 4 and Group 5
- Group 8: All CICS systems in CICSplex 1

CICSplex 2

- Group 1: TOR 3
- Group 2: AOR 6 through AOR 9
- Group 4: All CICS systems on System E
- Group 5: All CICS systems on System G
- Group 6: Group 4 and Group 5
- Group 7: All CICS systems in CICSplex 2

Notice that Group 7 in CICSplex 1 and Group 6 in CICSplex 2 comprise of other groups. Defining groups within groups is very efficient, both for you (because it means less effort) and for CICSplex SM.

Group 8 in CICSplex 1 and Group 7 in CICSplex 2 include the same set of CICS systems as the CICSplex to which it belongs. These are often useful groups to define because the scope value (as specified for a monitor specification, for example) can be a CICS system or a CICS system group name only: it cannot be the name of a CICSplex.

This is merely an initial list of system groups. It is likely to be added to (or altered) when BAS, WLM, RTA, and monitoring requirements are identified.

You can identify one or more subsets of the CICS systems within a CICSplex as a CICS system group, which can be manipulated as a single entity and independently of the rest of CICSplex. For example, if you define a CICSplex made up of TOR, AOR, and FOR CICS systems, you might want to define the AORs as a CICS system group, so that you can use a single CICSplex SM command to make changes to, or request data from, all CICS systems in that category.

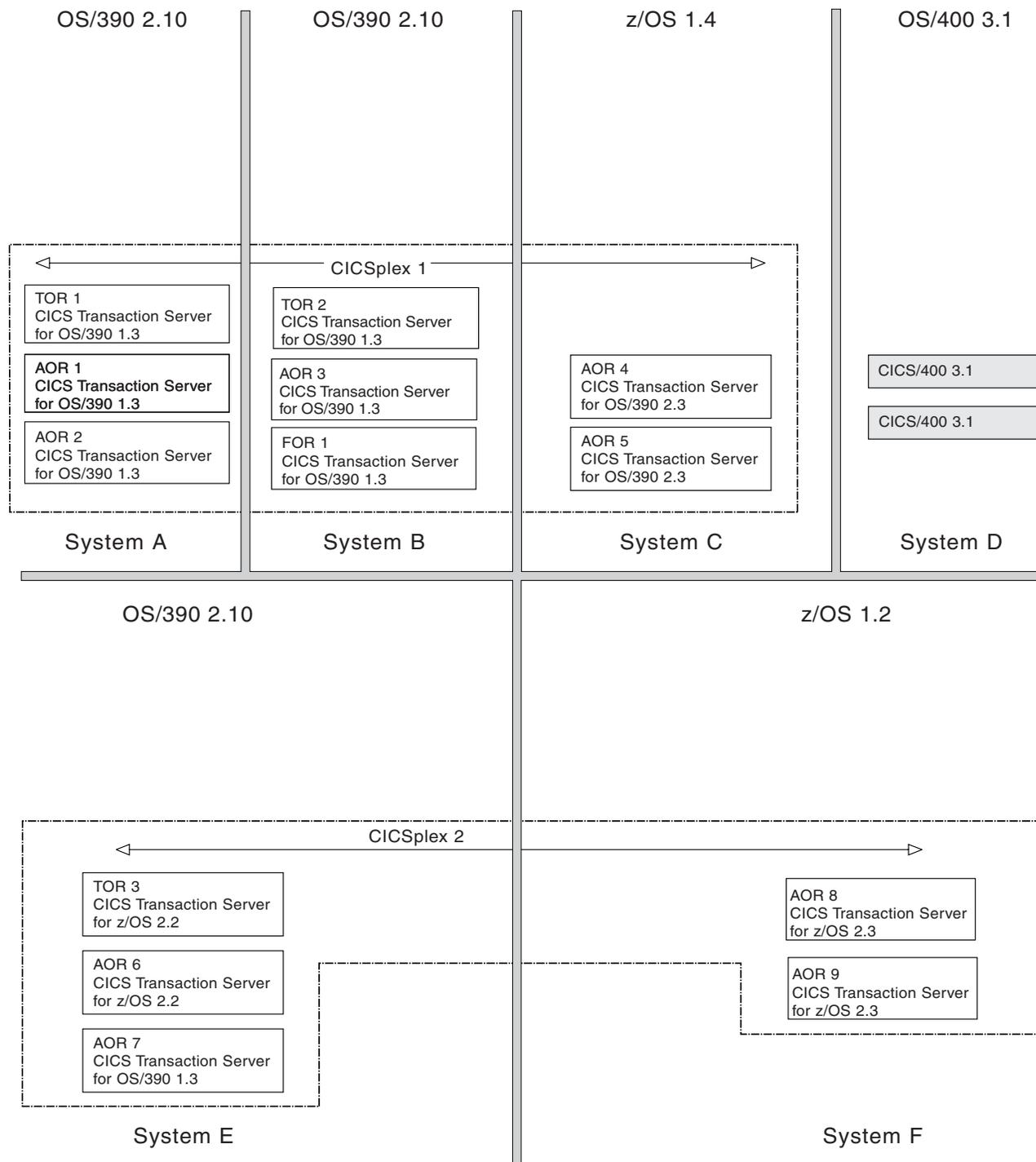


Figure 5. Identifying the CICSplexes. The example enterprise operates as two discrete units: the first three MVS images process a different workload from the remaining two, and there is no sharing of resources between those two groups. Accordingly, the enterprise CICS systems are to be divided between two CICSplexes. CICS systems that cannot be managed by CICSplex SM (the CICS/400 systems, in the example) have been shaded.

Alternatively, you could define a single group for any of the following:

- Particularly heavily loaded CICS systems
- CICS systems that have different security requirements from the other CICS systems in the CICSplex
- CICS systems in which particular applications run

CICS system groups, unlike CICSplexes, do not have to be mutually exclusive: a CICS system can belong to any number of groups within a CICSplex. However, because the CICS system group is a subset of the CICSplex, a system group cannot cross CICSplex boundaries.

Groups within groups

You can create CICS system groups from other groups. For example, if you want a single group to contain all AORs and all TORs in CICSplex, you can define its members as:

- The CICS system group comprising all AORs
- The CICS system group comprising all TORs

Any duplication of CICS system names that occurs in this way (for example, if a particular CICS system belongs to more than one constituent group) is accommodated by CICSplex SM. When a CICS system group is the target of a CICSplex SM command, CICS systems appearing in the group more than once are acted on once only.

Locating CMASs

The CMAS is the component of the CICSplex SM topology that is responsible for most of the work involved in managing and reporting on systems and their resources. It is the CMAS that is responsible for presenting the SSI to the operator. Each CICSplex is managed by at least one CMAS. This section gives you guidance on deciding where to put CMASs and how many to have.

Where to install CMASs

Now that you have decided which CICS systems are to be managed by CICSplex SM, and how they are to be organized into CICSplexes, you should think about where CMASs are required. The rules and recommendations governing where a CMAS can, and must, be installed are:

- Each CICSplex must be managed by at least one CMAS.
- Each CICSplex must have a maintenance point CMAS.
- A CMAS can participate in the management of multiple CICSplexes.
- The CMAS, which is itself a CICS system, must be a CICS Transaction Server for OS/390, Version 1 Release 3 (or later) system. If you are running CICS systems that are not directly-connectable to CICS Transaction Server for z/OS, Version 3 Release 1, the CMAS must be CICS Transaction Server for OS/390, Version 1 Release 3 or later, and appropriate to the level of CICSplex SM that you are running. See “CICS system connectivity” on page 2 and the *CICS Transaction Server for z/OS Migration from CICS TS Version 2.3*
- If you want CICSplex SM to issue SNA generic alerts to a NetView instance, a CMAS must be installed on the same MVS image as the NetView instance.
- It's usual to have one CMAS on each MVS image on which managed CICS systems are running.
- You can have more than one production CMAS in an MVS image, but it is unlikely to be necessary.

You should also consider the performance implications of your decision. Each CMAS has its space requirements and its own data repository. If you plan to use the ISPF EUI on the image, you will need to install a CAS. For details of estimated space requirements of the CICSplex SM components, see Chapter 9, “Planning CICSplex SM installation and setup,” on page 91.

If we apply these rules and recommendations to the example enterprise, you can see that:

- You must have at least one CMAS in the enterprise.
-
- NetView is installed on System C; to send SNA alerts to that NetView instance, you must install a CMAS on the same MVS image.

Figure 6 on page 26 shows the example map updated again, this time to show the application of these CMAS rules and recommendations.

CMAS-to-CMAS links

When a CICSplex crosses multiple MVS images, multiple CMASs are likely to be involved in managing that CICSplex. The CMASs must be connected to each other (using CICS intercommunication methods) so that data relating to the CICS systems belonging to the CICSplex is accessible by all relevant CMASs, and an SSI can be presented to the CICS operator. The minimum requirement is that CMASs managing the same CICSplexes be linked to form an open chain of CMASs, so that each CMAS is linked, if only indirectly, to every other CMAS. Meeting this requirement ensures that there is at least one path from each CMAS to every other CMAS involved in managing the same CICSplex. To achieve the best performance and availability, link every CMAS directly to every other CMAS.

Figure 7 on page 27 illustrates the concept of direct and indirect CMAS-to-CMAS links.

The links between multiple CMASs involved in managing the same CICSplex are responsible for the SSI of the CICS systems or systems that belong to that CICSplex. However, you do have a degree of choice regarding the *number* of links you define. First, here's a reminder of the rules and recommendations concerning CMAS-to-CMAS links:

- To establish an SSI of a CICSplex, the CMASs managing that CICSplex should at the very least be linked to form an open chain.

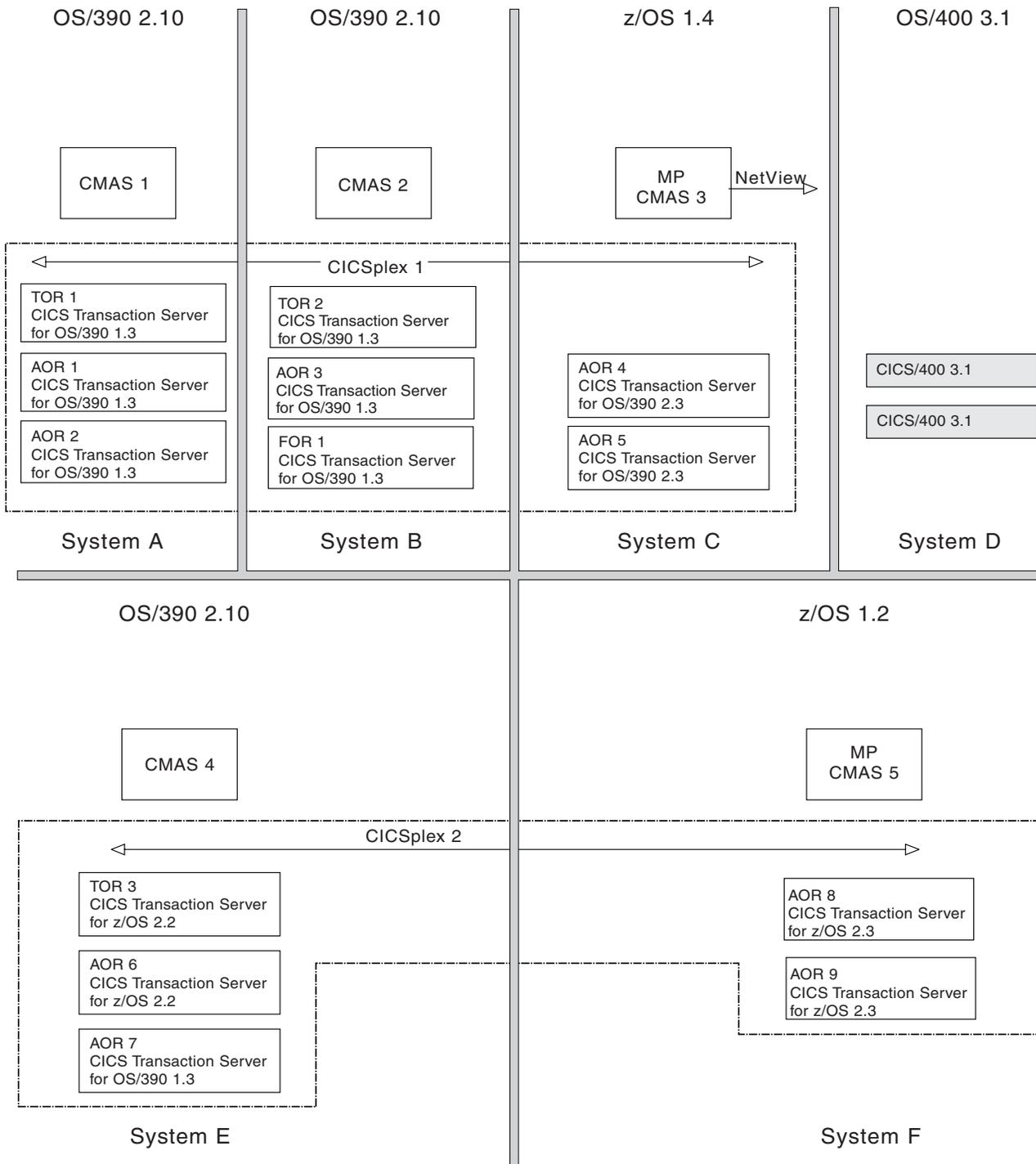


Figure 6. Adding CMASs to the enterprise map. A CMAS has been installed on each MVS image. CMAS 3 can issue SNA generic alerts to the NetView instance on the same MVS image. CMAS 3 is the maintenance point CMAS for CICSplex 1, and CMAS 5 is the maintenance point CMAS for CICSplex 2. (That is, CMAS 3 will be the context CMAS when you define CICSplex 1, and CMAS 5 will be the context CMAS when you define CICSplex 2 to CICSplex SM.)

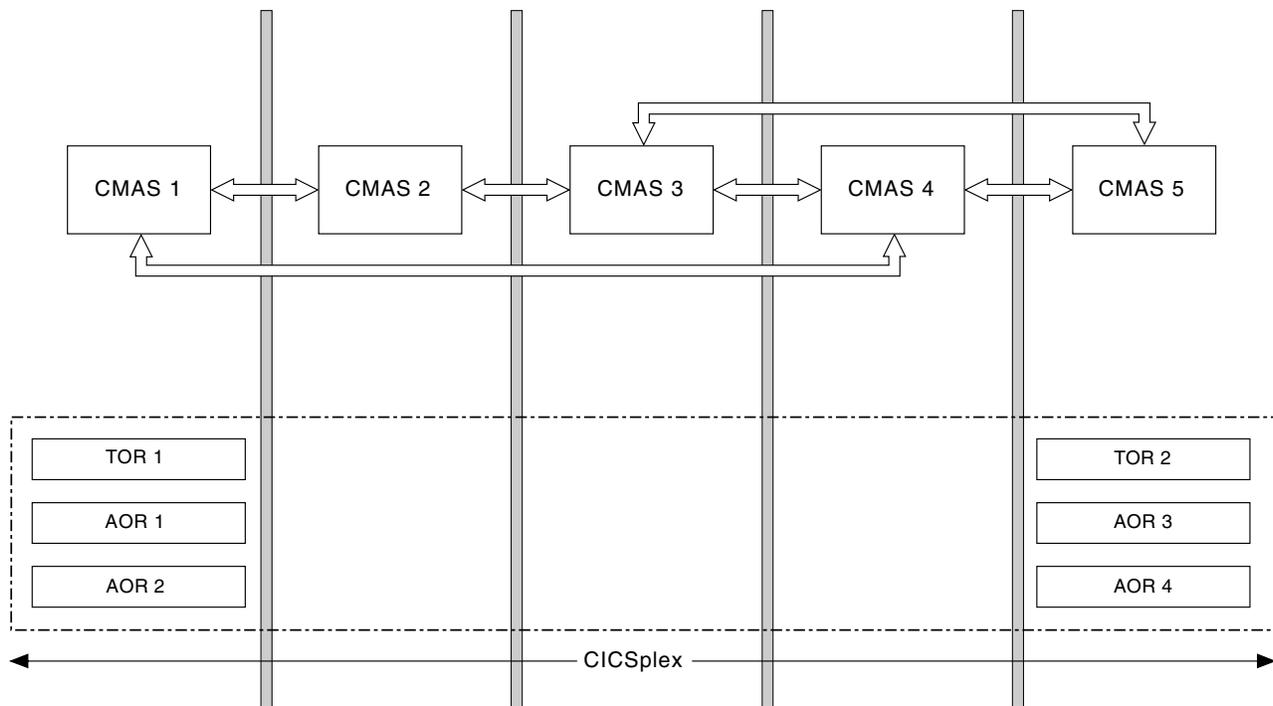


Figure 7. CMAS-to-CMAS links. This CICSplex is made up of six CICS systems, three managed by CMAS1 and three managed by CMAS5. In the absence of a direct link between the two CMASs, CICSplex SM is able to navigate dynamically around its “network” to gather information relevant to the CICSplex. It could go via CMAS4 or, if CMAS4 is unavailable, for example, it could go via CMAS2 and CMAS3. For optimum performance, however, a direct link should be added between CMAS1 and CMAS5.

- The more links you establish, the better the performance. The best performance is achieved when every CMAS is directly connected to every other CMAS involved in the management of a single CICSplex.
- A multiplicity of CMAS-to-CMAS links is also better for availability: if CMAS 1 is at the end of the chain and is connected to CMAS 2 only, the failure of CMAS 2 breaks the chain and leaves no connection to CMAS 1.

In the example configuration, full CMAS-to-CMAS connectivity is established for each of the two CICSplexes. That is, for management of CICSplex 1, CMASs 1, 2, and 3 are directly connected to each other; for management of CICSplex 2, CMASs 4 and 5 are directly connected to each other. Thus, eight CMAS-to-CMAS links must be defined:

| | |
|-------------------|-------------------|
| <i>CICSplex 1</i> | <i>CICSplex 2</i> |
| CMAS1 to CMAS2 | CMAS4 to CMAS5 |
| CMAS1 to CMAS3 | CMAS5 to CMAS4 |
| CMAS2 to CMAS1 | |
| CMAS2 to CMAS3 | |
| CMAS3 to CMAS2 | |
| CMAS3 to CMAS1 | |

You could have decided not to link CMAS 1 to CMAS 3 directly. This arrangement would have met the minimum CMAS-to-CMAS link requirement (by leaving us with an open chain of CMASs) and would have reduced the number of links to be defined by two. Had you done so, information requested from System C about CICS resources on System A, for example, would be obtained indirectly (via the

adjacent CMAS, CMAS 2) rather than directly. Wherever full connectivity is not established among a group of CMASs, CICSplex SM works out the quickest route to the requested data dynamically.

The performance impact is slightly higher when information is obtained indirectly, but this needs to be set against the overhead of setting up and maintaining the CMAS-to-CMAS links. In fact, even if you had defined one CICSplex rather than two in your example, full connectivity among the CMASs would have been feasible. (That is, you would have needed 20 CMAS-to-CMAS links, because the number of links required is $n^2 - n$, where n is the number of CMASs to be connected.) However, if you had ten CMASs managing a single CICSplex, the number of links required would jump to 90; if you had 15 CMASs, the number would be 210. In summary, you have to decide how many direct links you are prepared to define, and where indirect links can be accommodated, always remembering that the minimum requirement is for an open chain of CMAS-to-CMAS links.

Figure 8 on page 29 shows the CMAS-to-CMAS links in the example configuration.

When there are multiple CMASs on the same MVS system, managing the same CICSplex, a local MAS that specifies only the CICSplex(name) CICSplex SM system parameter, and not the CMASYSID(name) CICSplex SM system parameter, will connect to the last CMAS to initialize on the MVS image that manages the named CICSplex for a specific release of CICSplex SM. See the *CICS Transaction Server for z/OS Installation Guide* for a description of CICSplex SM system parameters.

CMAS-to-CMAS links for multiple CICSplexes

In our example, there are two CICSplexes. So that the CICSplex SM operator can access CICSplex 2 data from System B, for example, you could establish a link between any one (or more) of the CMASs involved in managing CICSplex 1 and any one (or more) of the CMASs involved in managing CICSplex 2. For example, you could establish a link from CMAS 1 to CMAS 5, thereby forming a chain of all of the enterprise CMASs.

Alternatively, data belonging to one CICSplex can be accessed from an MVS image on which a second CICSplex resides via CAS-to-CAS links. With the configuration you have chosen, CICSplex SM information about an AOR on System E, for example, requested from the CICSplex SM user interface on System A, can be obtained via CAS-to-CAS links rather than CMAS-to-CMAS links. CAS-to-CAS links are discussed in “CAS-to-CAS links” on page 31.

Locating the maintenance point CMAS

When a single CICSplex is managed by multiple CMASs, one of them is designated the *maintenance point* CMAS. The maintenance point CMAS is responsible for maintaining all CICSplex SM definitions relating to a CICSplex, and for distributing up-to-date information to the other CMASs involved in managing the same CICSplex, so that no data repository is out of step with any other.

The maintenance point CMAS is the CMAS that is the *context* value when the CICSplex is defined to CICSplex SM. (The context is described on page “Display formats” on page 12.) Each CICSplex can have only one maintenance point CMAS, but a single CMAS can be the maintenance point for more than one CICSplex. If the maintenance point CMAS is unavailable at any time, changes to all

CICSplex SM functions except BAS can be made at another CMAS managing the CICSplex. This CMAS becomes the *temporary maintenance point CMAS*. Any changes are temporary.

BAS does not support temporary maintenance point CMASs. The maintenance point CMAS must be available if you are creating or maintaining BAS definitions.

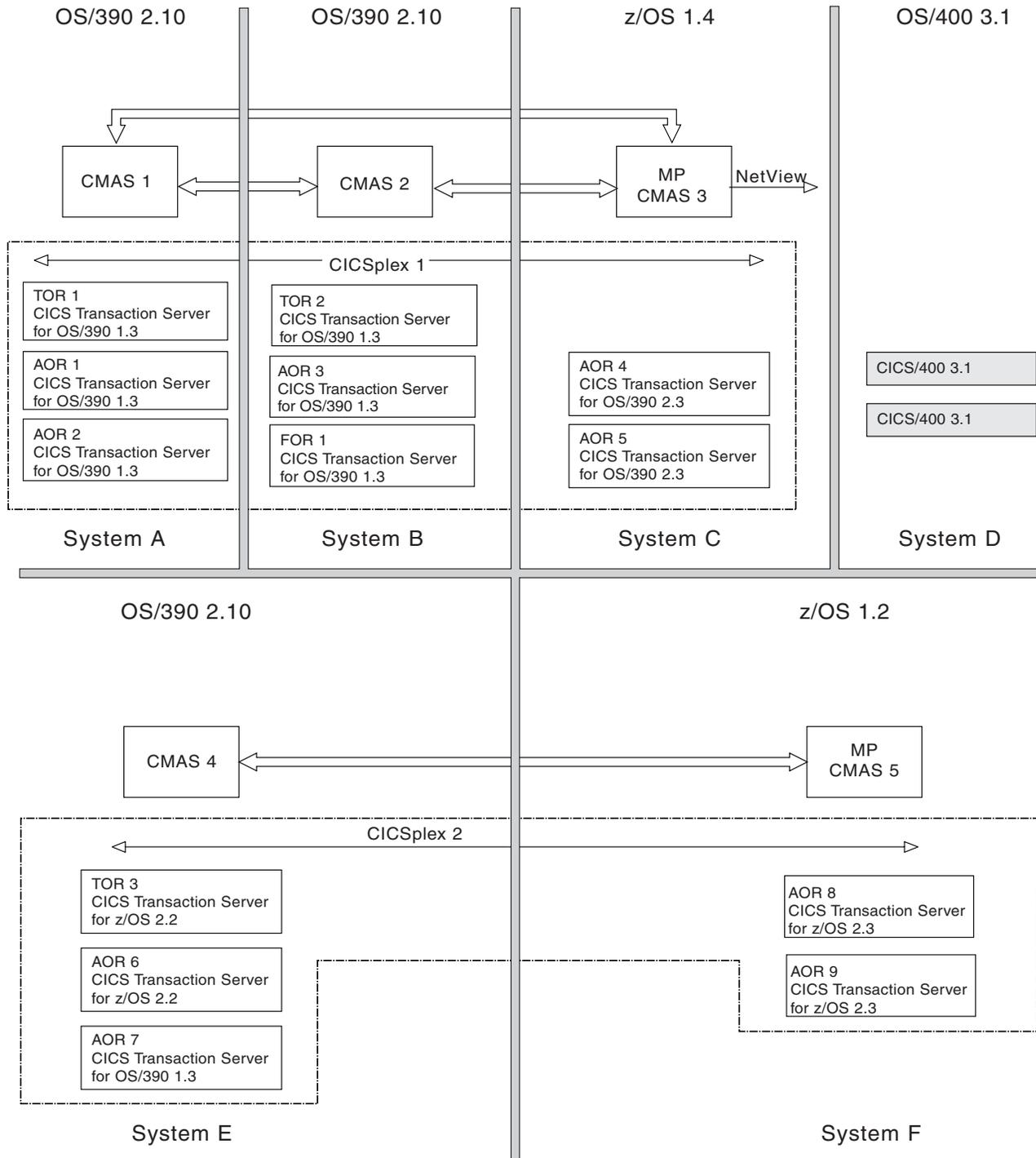


Figure 8. Adding CMAS-to-CMAS links to the example map. Eight links have been defined, giving full connectivity among CMASs managing the same CICSplexes.

A maintenance point CMAS can also function as a normal (not maintenance point) CMAS for other CICSplexes.

Note: CICSplex SM's single point of control means that changes to the maintenance point CMAS can be made from any system on which the CICSplex SM EUI is available.

Locating CASs

Once you have decided where to install CMASs and how to link them, you can turn your attention to the CASs.

Where CASs are required

The rules and recommendations on where to locate a CAS are:

- To set up the CICSplex SM component infrastructure, you need a CAS on one of your MVS images. This CAS can be used to support the MVS/TSO ISPF user interface used to set up CICSplex SM.
- There must be one CAS on each MVS image from which you plan to use the MVS/TSO ISPF user interface to CICSplex SM. You cannot access CICSplex SM data *directly* from an MVS image that doesn't have a CAS. For example, CICSplex SM data cannot be accessed from the System B shown in Figure 9, because the image does not have an MVS/TSO ISPF EUI, even though managed CICS systems are running there. This does not mean that the CICSplex SM data cannot be accessed at all, but that it must be accessed from other MVS images.
- You can install a CAS on an MVS image on which there is no CMAS. Provided that the MVS image has the required level of ISPF installed, it can be used as an access point for CICSplex SM data on other MVS images.
- If a CMAS only participates in management of MASs in the CICSplex, and you do not want to set its name as a context (for example, if it is a maintenance point), you do not need a CAS on the same MVS image.
- Although the CAS supports the MVS/TSO ISPF EUI, TSO itself doesn't have to be available on every MVS image on which you have a CAS. The absence of MVS/TSO from a particular image simply means that the user cannot access CICSplex SM data from that image. It can, however, be accessed from other images. Figure 9 on page 31 illustrates three valid CAS scenarios.

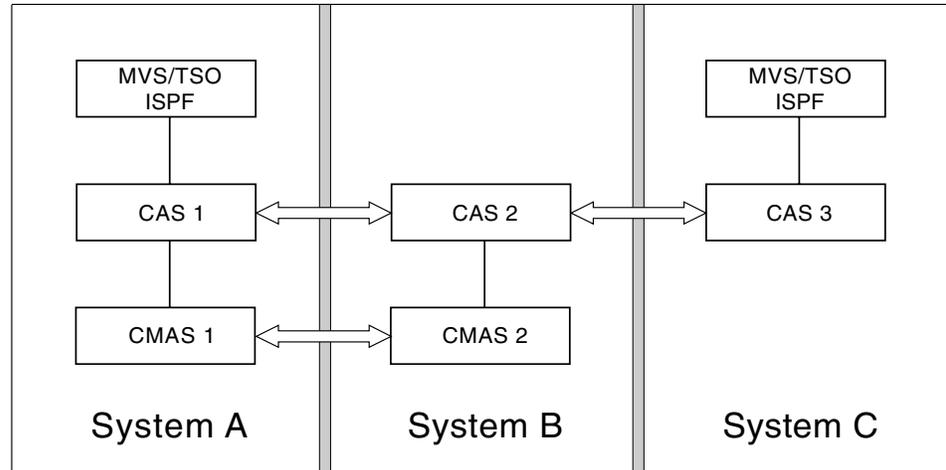


Figure 9. Valid CAS configurations. The most likely configuration is that shown on System A. However, those shown on System B and System C are also valid. CICSplex SM data cannot be accessed from System B because there is no MVS/TSO instance. However, because CAS2 has links to other CASs on MVS images where MVS/TSO is available, the data on System B is accessible from other MVS images. System C has no CMASs, but can still be used as an access point for CICSplex SM data on other MVS images.

|
|
|

It's very easy to update the example map by applying these rules: there are five CMASs, so there must be five CASs, one on each of the MVS images on which a CMAS is installed. Figure 10 on page 32 shows the addition of the CASs..

CAS-to-CAS links

If you define no CAS-to-CAS links, the CICSplex SM operator can sign on to CICSplex SM via CAS 1 (or CAS 2, or CAS 3) and still have a complete view of the CICS systems and resources in CICSplex 1 because the CMASs managing CICSplex 1 are linked to each other. The same observation is true of CICSplex 2. For example, a CICSplex SM operator accessing CICSplex SM via CAS 5 can see data for TOR 3 via the links between CMAS 4 and CMAS 5. However, for both performance and availability reasons, it is better to establish links between CASs also.

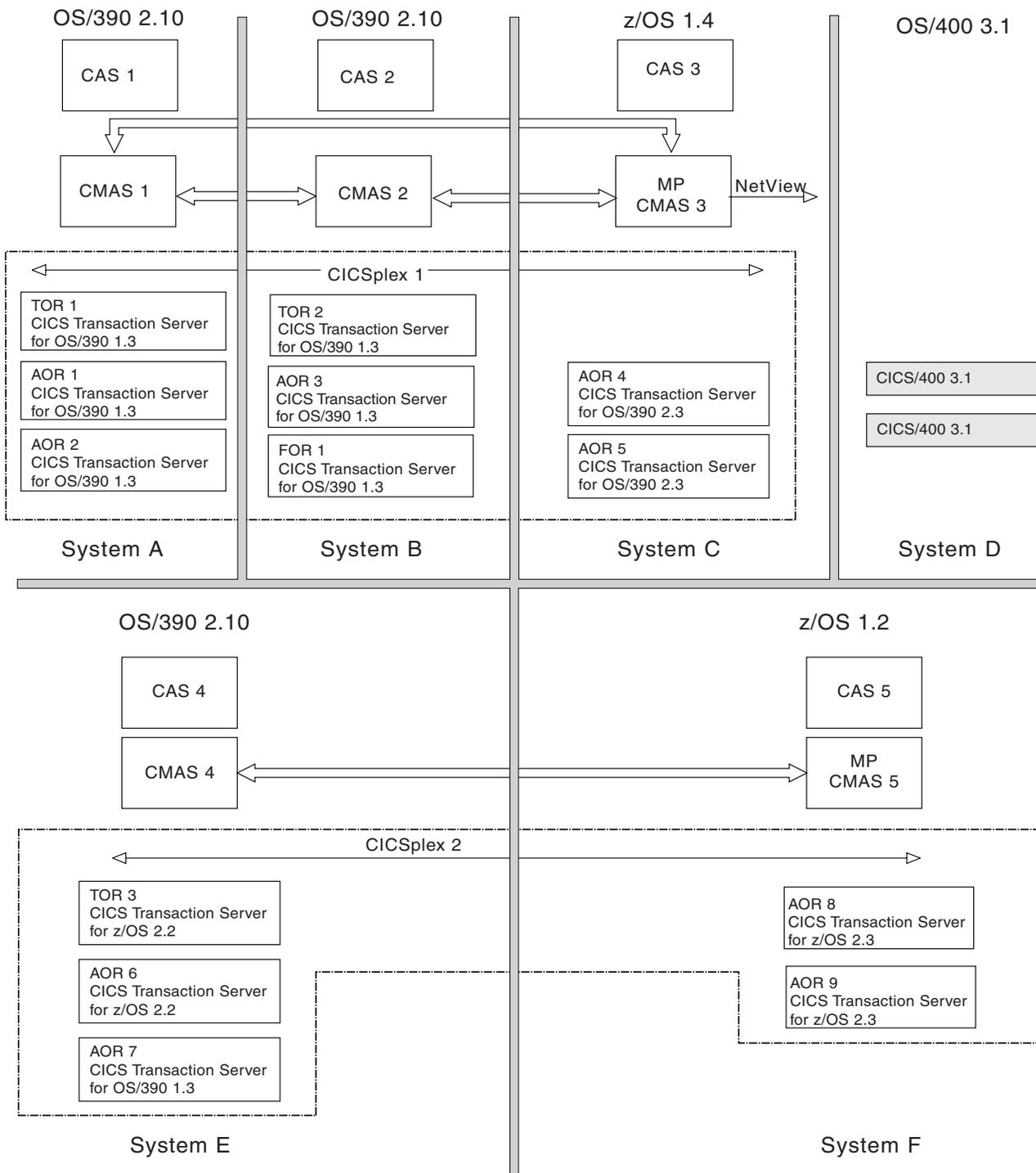


Figure 10. Adding CASs to the example map. A CAS has been added to every MVS image to which a CMAS has been assigned.

A second consideration in the example is the availability of a single point of control for the enterprise. If you add no CAS-to-CAS links, a CICSplex SM operator cannot see CICSplex SM data for CICSplex 2 from MVS image 3, for example. This arrangement might be suitable in some enterprises (indeed, it might be the reason why it was decided to have multiple CICSplexes in the first place). However, if you do want a single point of control for the enterprise and have multiple CICSplexes, CAS-to-CAS links are required.

Thus, CAS-to-CAS links can provide a single point of control for the enterprise by allowing the CICSplex SM operator access to CMASs that may not be even indirectly connected to each other; CMAS-to-CMAS links provide an SSI for all MASs that connect to those CMASs; and CICSplex SM exploits a combination of CAS-to-CAS links and CMAS-to-CMAS links to give the best possible performance.

In the example, few CAS-to-CAS links are essential. If you add two links, one from CAS 1 to CAS 4, and one from CAS 4 to CAS 1, both CAS 1 and CAS 4 can function as a single point of control because each has access to data about CICSplexes 1 and 2. However, for maximum performance and availability, and because the total number of CASs in the enterprise is small, you are going to link every CAS to every other CAS. As with the CMAS-to-CMAS links, the formula for calculating the number of links to be defined is $n^2 - n$, where n is the number of CASs. In the example, there are five CASs, and so 20 links must be defined to establish full connectivity for the enterprise. If you have room on your map, add the CAS-to-CAS links. As the example map is beginning to look crowded, the CAS-to-CAS links are listed separately in Table 2.

Table 2. CAS-to-CAS links required for full connectivity among the example CASs

| CAS 1 to: | CAS 2 to: | CAS 3 to: | CAS 4 to: | CAS 5 to: |
|-----------|-----------|-----------|-----------|-----------|
| CAS2 | CAS1 | CAS1 | CAS1 | CAS1 |
| CAS3 | CAS3 | CAS2 | CAS2 | CAS2 |
| CAS4 | CAS4 | CAS4 | CAS3 | CAS3 |
| CAS5 | CAS5 | CAS5 | CAS5 | CAS4 |

Full connectivity among the enterprise CASs is recommended for performance and availability reasons. However, it is not essential: if the enterprise CMASs are adequately connected to each other, a single point of control for the enterprise can be established without full connectivity among the CASs.

Naming your CICSplex SM entities

All the entities that you have defined in this chapter, and entered on to your CICSplex SM map, have to be named. You must devise a convention for naming those entities that is both meaningful and extendable. CICSplex SM supports the use of generic names in many of its commands. For example, you can specify that all transactions whose names begin with the letters “DNW” are to be monitored once every 300 seconds. You do not have to name each transaction individually.

The rules governing use of generic names are as follows:

- The asterisk character (*) can be used in place of one or more characters. It can appear once only in any name, and must appear at the end of any partial string. For example, the string “DNW*” means all names beginning with the letters DNW.
- The plus sign (+) can replace any single character and can occur in any position in the name. For example, “DNW+++L” means any name of 7 characters starting with the letters DNW and ending with the letter L.

A staged implementation

If you have a very large number of CICS systems, or if you have decided to create multiple CICSplexes, you might want to consider implementing CICSplex SM in stages. You can install CICSplex SM on a subset of your systems and use it to manage a single CICSplex or, if you have defined a single CICSplex for the enterprise, you can use it to manage particular system groups. Additional CICS systems can be added to the configuration gradually.

If you opt for the gradual approach, you will need to revisit your final map of the enterprise CICS systems and identify those you plan to start with. For example, if a staged implementation were adopted for your example map, you could decide to begin by implementing CICSplex 2 only. The CAS-to-CAS links and the CMAS-to-CMAS links would be much reduced in number (two CAS-to-CAS links rather than 20, and two CMAS-to-CMAS links, rather than 12), but the other decisions made about CICSplex 2 are still valid. That is, the same CICS system groups can be defined, and the CAS and CMAS installations are still required.

Updating the initial design

When you have been working with your initial CICSplex SM configuration for a while, it's possible that you will want to do some fine-tuning. For example, it might become apparent that you need some additional CMAS-to-CMAS links, or some extra CICS system group definitions. You will also need to make changes as more CICS systems or regions are created, or if you implement the initial configuration in stages. Whatever the reason for updating the initial design, you are recommended to keep the map of the CICS systems in your enterprise up-to-date, because an accurate map will make it easy for you to maintain the CICSplex SM configuration and topology data.

Where next?

Now that you have a map of your enterprise, you can start the planning for the installation of CICSplex SM and the creation of your proposed configuration. See Chapter 9, "Planning CICSplex SM installation and setup," on page 91 for guidance on planning for the installation of CICSplex SM.

Chapter 4. Defining the CICSplex SM configuration and topology

Once CICSplex SM has been installed, and you have a map of your CICSplex SM environment, you can define the configuration and topology. This chapter covers:

- “Defining your CICSplex SM configuration”
- “Defining CICSplex topology” on page 37
- “Where next?” on page 40.

Defining your CICSplex SM configuration

For your CICSplex SM configuration, you need to define:

- CAS-to-CAS links
- The CMAS configuration

Defining CAS-to-CAS links

CAS-to-CAS links are defined using the PlexManager service utility. You can use PlexManager also to determine the status of the connections between a CAS and its associated CICSplex SM address spaces (CMASs) and CICSplexes, and to manage the connections between CASs.

Defining CAS-to-CAS links is one step in the process of installing a CAS, which forms part of the setup procedure. This is described in detail in *CICSplex System Manager Administration*.

Managing your CAS environment

You can use the following PlexManager views to manage your CAS environment:

CASACT

Manage the activity of your VTAM cross-system communication links.

CASDEF

Create and maintain CAS cross-system communication definitions.

CASDEFD

Display information about a specific CAS definition.

PLEX Display a list of the available combinations of product and context (or *service points*) in one or more systems known to the CAS to which you are currently connected.

PLEXOVER

Display a list, by context, of the available service points known to the CAS to which you are currently connected.

SCREENS

Display a list of user-created screen configurations.

Defining the CMAS configuration

For each CMAS you have defined, you must provide the following information to CICSplex SM:

- The name of each CICSplex that it is to manage, if it is a maintenance-point CMAS
- Link definitions for each CMAS to which it is connected

CMAS configuration definitions

To enter this information, you can use the appropriate CMAS Configuration Administration (ADMCONFIG) views, or the batched repository-update facility.

Managing the CMAS configuration

You use the following views to manage your CMAS configuration:

BATCHREP

To create, update, remove, list, or dump one or more definitions from the data repository associated with the local CMAS.

CMTCMDEF

To display information about the direct LU6.2 and MRO communication links between the local CMAS and any other CMASs.

CPLEXDEF

To define a CICSplex to the local CMAS and to administer CICSplex definitions for the local CMAS.

CPLXCMAS

To administer information about CMASs associated with a CICSplex, where the local CMAS is the maintenance point for that CICSplex.

CICSplex SM also provides the following views that you can use to manage CMAS configuration definitions when the associated CMASs are active.

CICSPLEX

To display information about the CMASs associated with a CICSplex known to the local CMAS.

CMAS

To display information about CMASs known to the local CMAS, shut down a CMAS, and alter CMAS component trace settings.

CMASD

To display information about a specific CMAS known to the local CMAS, shut down a CMAS, and alter CMAS component trace settings.

CMASPLEX

To display information about the CICSplexes being managed by the local CMAS.

CMTCMLNK

To display information about all or specific CMASs linked to the local CMAS and to discard a CMAS-to-CMAS link.

CMTCMLND

To display information about the link between a specific CMAS and the local CMAS and to discard the CMAS-to-CMAS link.

CMTPMLNK

To display information about all or specific MASs linked to the local CMAS and to discard the CMAS-to-MAS link.

CMTPMLND

To display information about a specific MAS linked to the local CMAS and to discard the CMAS-to-MAS link.

See *CICSplex System Manager Administration* for details of all these views.

Defining CICSplex topology

When you have identified your CICSplexes and MASs, you need to create the definition objects that will identify them to CICSplex SM. You should be aware that you will not have all the information required at this point; more information will emerge as the design of your CICSplex SM system proceeds.

More information about designing CICSplexes is given in “Designing your CICSplexes” on page 17

Preparing CICSplex definitions

Each CICSplex must be defined to CICSplex SM by a name that is unique within the enterprise. Naming conventions are given in “Naming your CICSplex SM entities” on page 33. You also should be aware that you will need other information for the CICSplex definition, which will become available as you proceed with the design of your CICSplex SM system. This information includes:

- The name of the CMAS to which the CICSplex is to be defined. (This is the maintenance point CMAS for the CICSplex.)
- The names of any other CMASs that are to participate in management of the CICSplex.
- A time zone for the CICSplex SM monitor function.
- Whether CICS command checking is to be simulated.
- Whether CICS resource checking is to be simulated.

Managing the CICSplex definitions

A CICSplex is identified to CICSplex SM via the CPLEXDEF view, which is described in *CICSplex System Manager Administration*.

To enter this information, you can use the appropriate CMAS Configuration Administration (ADMCONFIG) views, or the batched repository-update facility.

When you have created a CICSplex definition, you can associate CICS systems (MASs) and CICS system groups with the CICSplex. See “Preparing CICS system definitions.”

Preparing CICS system definitions

In order to be managed by CICSplex SM, a CICS system must be associated with a CICSplex that is defined to CICSplex SM. The definition establishing this association also contains information about how the CICS system is to use the BAS, WLM, RTA, and monitoring components of CICSplex SM.

To create CICS system definitions, and to display information about CICS systems defined to a CICSplex, you use the CICSSYS view.

For every CICS system that is to be managed by CICSplex SM, you must gather the following information:

- The name by which this system is to be known to CICSplex SM
- The name of the CICSplex to which the CICS system is to belong
- The APPLID, that is, the VTAM application identifier for the CICS system
- The SYSID, that is, the CICS system identifier

This is sufficient data for adding the MAS definition to the CICSplex SM data repository. If you know that you have special BAS, WLM, RTA, or monitoring requirements for any CICS system, you can enter those at the same time.

CMAS configuration definitions

Alternatively, you can update the CICS system definition later as new requirements become apparent. For example, you might want to collect some of the following information, which is optional and can be supplied after the CICS system is defined to CICSplex SM:

- If system availability monitoring (SAM) is to be implemented, the name of the primary CMAS, which is the CMAS to which the CICS system usually connects, and the hours of operation of the CICS system
- Whether CICS command checking is to be simulated
- Whether CICS resource checking is to be simulated
- The time zone of the CICS system, if it is not to be the same as that of its CMAS

To enter this information, you can use the Topology Administration (ADMTOPOL) views, or the batched repository-update facility. Note that the ADMTOPOL views are also used to create time-period definitions.

Preparing CICS system group definitions

For each CICS system group to be defined to CICSplex SM, you need provide only a name for the group, which may be unique within the CICSplex. The CICS system group must be defined before you can add CICS systems to it. To define a CICS system group, you use the ADMTOPOL views.

Defining CICS system groups, perhaps more than any other part of the design stage, is an iterative process and a cumulative one. It might be obvious to you already that there are natural groupings of CICS systems within your CICSplexes, and that you will want to manipulate each of those groups as a single entity. If this is the case, start to document the groups now. Because groups do not have to be mutually exclusive, and because a CICS system can belong to any number of groups, you can add CICS system group definitions as new requirements emerge.

To create CICS system groups, you use the CICSGRP view.

You can display information about the CICS system groups, and the CICS systems associated with them, that are known to the CICSplex identified as the current context, by using the SYSGRPC view.

CICS system link definitions

For a CICS system to communicate with another CICS system, the connection between the two systems may be specified by a *link definition*. A link definition is used to automatically create the matching CICS resource definitions (connection and session definitions) required in the two connected CICS systems. These link definitions are used by CICS (not CICSplex SM); for example, for transaction routing between CICS systems.

To create CICS system link definitions, and to display information about those definitions, you use the SYSLINK view.

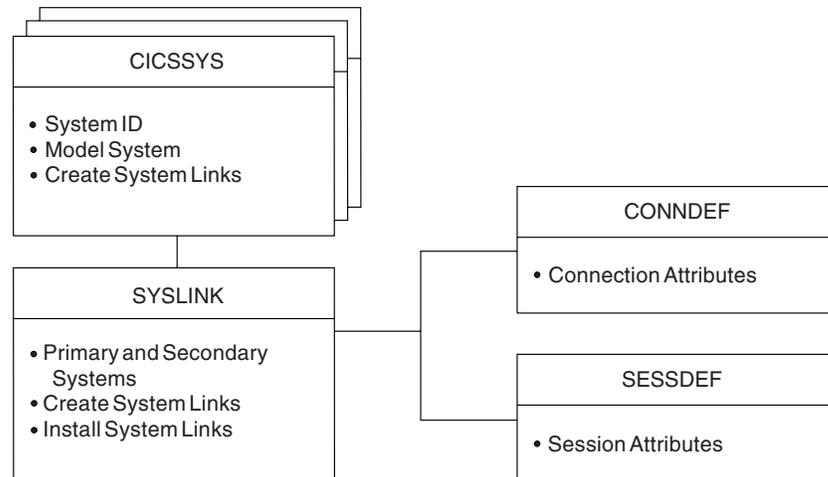


Figure 11. CICS system link definitions

Managing CICS system definitions

CICS system definitions are managed through the following objects:

CICSGRP

To create and administer CICS system groups within a CICSplex, including adding CICS systems to CICS system groups. This view is described in *CICSplex System Manager Administration*.

CONNDEF

To create and administer connection definitions that identify remote systems that a CICS system communicates with using intersystem communication (ISC) or multiple region operation (MRO). This view is described in *CICSplex System Manager Managing Business Applications*.

CICSSYS

To create and administer the definition that associates a CICS system with a CICSplex. The definition also contains information about how the CICS system is to use the WLM, real-time analysis, resource monitoring, and business application services components of CICSplex SM. This view is described in *CICSplex System Manager Administration*.

PERIODEF

To create and administer time period definitions, which identify specific ranges of hours and minutes to be used with resource monitoring and real-time analysis. This view is described in *CICSplex System Manager Administration*.

SESSDEF

To create and administer session definitions that describe the nature of logical links between systems that communicate using intersystem communication (ISC) or multiple region operation (MRO). This view is described in *CICSplex System Manager Managing Business Applications*.

SYSGRPC

To monitor information about the CICS system groups, and the CICS systems associated with a CICSplex. This view is described in *CICSplex System Manager Administration*.

managing CICS system definitions

SYSLINK

To create and administer the links between CICS systems in the CICSplex. This view is described in *CICSplex System Manager Managing Business Applications*.

CICSplex SM provides the MAS view that you can use to manage configuration definitions when the associated CICS systems are active.

MAS To manage information about active CICS systems known to a CICSplex or connected to the designated CMAS involved in managing the CICSplex.

- Display information about the active CICS systems
- Browse or temporarily update the general, WLM, real-time analysis, resource monitoring, and MAS trace attributes of an active CICS system.
- Stop MAS agent code in an active CICS system.

This view is described in *CICSplex System Manager Administration*.

Note: Using the CICSSYS and PERIODEF views to update definitions affects both a currently running system and the definitions in the data repository. Updates made using the MAS view, however, are not included in the data repository.

Where next?

You have a working CICSplex SM environment, with your configuration and topology defined, and with the links between the entities defined. The entities can communicate with each other and you can run transactions. However, to optimize the operation of your enterprise and to take full advantage of the facilities offered by CICSplex SM, you need to set up your BAS, WLM, RTA, and monitoring requirements. For guidance, go to Chapter 5, “Managing resources using Business Application Services (BAS),” on page 41, Chapter 6, “Managing workloads,” on page 57, Chapter 7, “Exception reporting using real-time analysis (RTA),” on page 73, or Chapter 8, “Collecting statistics using CICSplex SM monitoring,” on page 83, as appropriate.

Chapter 5. Managing resources using Business Application Services (BAS)

This chapter covers:

- “What is Business Application Services (BAS)?”
- “Managing BAS” on page 45
- “Migrating your resource definitions” on page 48
- “Defining resources” on page 49
- “Using migration form BAS” on page 50
- “Using full-function form BAS” on page 50
- “Installing applications and resources” on page 52
- “Security considerations” on page 52
- “Planning for BAS” on page 53
- “Implementing BAS” on page 53
- “Where next?” on page 55

What is Business Application Services (BAS)?

Business Application Services (BAS) is an alternative to resource definition online (RDO) that enables you to manage CICS resources in terms of the business application to which they belong, rather than their physical location in the CICSplex. A business application can be any set of resources that represent a meaningful entity in your enterprise. Within the application, you can group together resources of a particular type and define the run-time characteristics of that group. At run-time, you can refine the selection of resources to be installed by using a filter expression, and you can change the resource attributes by using an override expression.

Limitations of RDO

RDO is the traditional CICS method of defining resources to CICS systems. With RDO, resource definitions are tied to a single group and groups that are processed sequentially from a group list. Any duplicate definition found later in the process overrides any earlier one. In addition, the very nature of the definitions that RDO requires and the extent to which they are available to multiple CICS systems can prove limiting. RDO has several disadvantages in a CICSplex SM environment:

- Resources are assigned to groups that are assigned to specified CICS systems.
- Resource definitions have to be duplicated across CICS systems, unless the CSD is shared.
- Each end of every communications link has to be defined explicitly.

BAS concepts

BAS enables you to view and manage your resources in terms of their use in your business, and independently of their physical location. This section introduces the basic concepts of the BAS function.

Logical scoping

Once your CICS resources are defined to CICSplex SM, you can monitor and control resources in terms of their participation in a named business application, rather than their physical location in the CICSplex. Logically-related resources can be identified and referred to as a set, regardless of where they actually reside at any given time.

Sets of definitions can be reused and associated with any number of other logical associations of resources that reflect your business needs, rather than your system configuration.

If you set the scope to be your application, any operation or monitoring views will display only those resources that satisfy your selection criteria. This gives you the power to control precisely how those resources are managed.

Multiple versions of a resource definition

With BAS, you can have multiple versions of the same resource. Each time you create a new definition for the same resource and the same name, BAS allocates a new version number. Note that a new version number is not created if you update the resource. You can specify a specific version of a resource definition by its version number.

Version support allows you to develop resource definitions as your business applications develop. You can then have, for example, a single version of a resource in multiple groups, or multiple versions of the resource throughout the CICSplex. Note that you can install only one version of a resource in a CICS system at one time.

Centralized resource definition

RDO definitions are held in a CICS System Definition (CSD) file. BAS resources are held on a central data repository that is accessed by all the CICS systems in the CICSplex.

The CICSplex SM data repository (EYUDREP) serves as the central repository for all your CICS resource definitions. This minimizes the number of resource definitions you need for your CICSplex by:

- Providing a single system image approach to defining CICS resources across all supported platforms.
- Producing both local and remote instances of a resource from a single definition.
- Managing multiple versions of a definition. For example, you can have different versions of the resources for an application as it progresses through a number of test phases.
- Generating multiple CICS communication links from a single set of connection and session definitions.

CICS system links

The traditional method of defining links between CICS systems is to use RDO to define connections and sessions between each CICS system. For each pair of CICS systems, there are four definitions: two connections and two sessions. Each connection and session is unique for a given pair of CICS systems.

| With BAS, communications links are created dynamically from model connection
| and session resource definitions. You define connections and sessions that describe
| the nature of the link. The BAS system link (SYSLINK) object is used to create the
| link between two CICS systems, by specifying the connection and session
| definitions to be used. The connection and session definitions can be used by any
| number of system link definitions that share the same characteristics. For more
| information see *CICSplex System Manager Managing Business Applications* and
| *CICSplex System Manager Administration*.

Distributed resource installation

Resources that are defined to CICSplex SM must still be installed in the appropriate systems, either by CICS or by CICSplex SM. You can use BAS to install your resources either automatically, at CICS initialization, or dynamically, while a system is running. A single resource can be installed in multiple CICS systems either locally or remotely, as appropriate.

Two forms of BAS

You can use BAS in two ways:

- Migration form, in which resource groups are associated with resource descriptions. In this form, resource descriptions are similar to group lists. See Figure 12 on page 44 and “Using migration form BAS” on page 50.
- Full-function form, in which resource assignments are used to qualify the contents of resource groups and resource descriptions, and to control the assignment of resources to CICS systems. See Figure 13 on page 44 and “Using full-function form BAS” on page 50.

You can use either or both of these forms, depending on the situation and degree of precision you require.

The simplest form is the migration form using resource descriptions. You create resource definitions and resource groups, and associate them with one or more resource descriptions. It is the resource description that defines the logical scope for the resources. The resources are assigned to specific CICS systems and you therefore still need separate definitions for a resource that is local to one system and remote to another. This approach to BAS is similar to using RDO, in that the resource description is analogous to the group list. You in effect create a version of your CSD on the CICSplex SM data repository. Though this is a good position from which to establish your CICSplex SM resources environment, you cannot take advantage of all the facilities offered by BAS.

To take advantage of the facilities offered by BAS to manage your resources in terms of the business application rather than location, you need to use resource assignments. A resource assignment selects resources of a particular type from a group, and assigns them to the appropriate CICS system. A resource assignment is associated with a resource description. The resource description then no longer functions like a group list, but becomes a user-defined, logical set of resources, such as an application.

Using resource assignments allows you to manage your individual resources and change their attributes for individual systems, or to suit special circumstances. You can:

- Control resources of a given type in a given group.
- Identify resources as either local or remote, and assign them to various CICS systems with a single resource definition.
- Process selected resources from a group by specifying a filter expression.
- Modify resource attributes for a particular use by specifying override expressions.

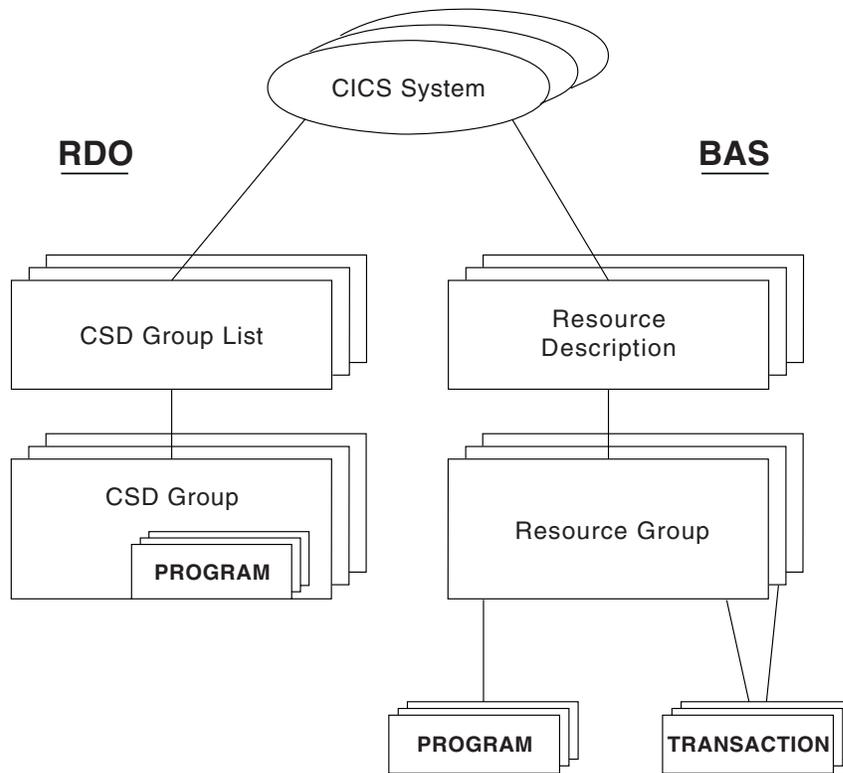


Figure 12. Migration form BAS

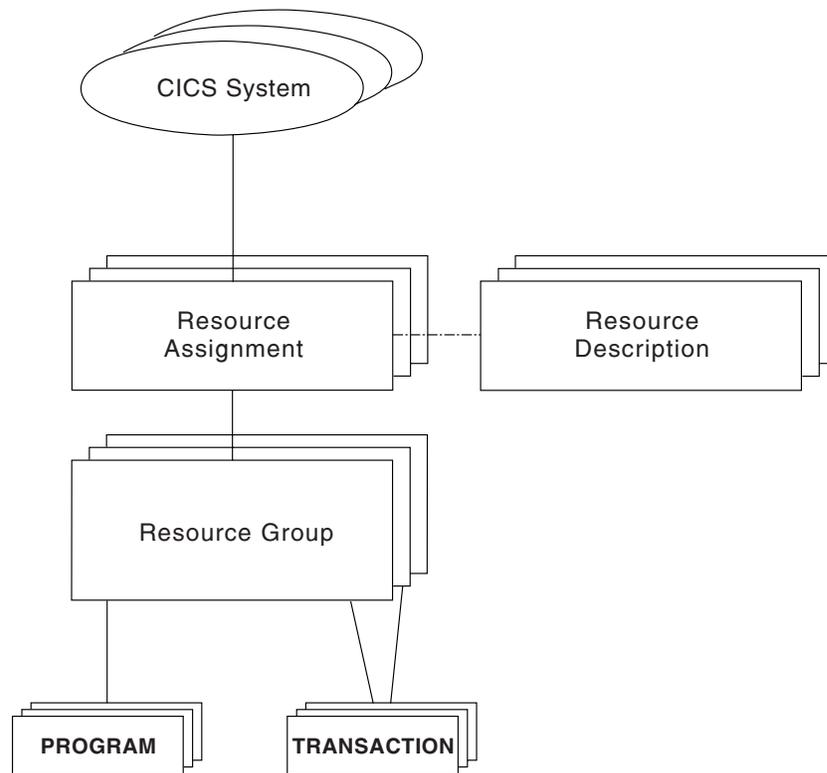


Figure 13. Full-function BAS

Advantages of using BAS

The BAS approach to the management of resources offers several advantages over RDO:

- A familiar, RDO-like definition process with a choice of interfaces (ISPF, batch or API).
- Logical scoping, that allows you to handle your resources in terms of business application rather than location.
- A common definition repository for all resources in a CICSplex, across all supported CICS platforms.
- A reduction in the number of definitions required, with the ability to reuse definitions and override individual attributes as needed.
- Consistent resource definitions and a great deal of control over the definition process.
- A resource can be added to additional regions simply by adding it to the groups that define where the application will run.
- Because the application is known to CICSplex SM, commands can be directed to a scope that matches the application, not to an arbitrary group of regions.

Managing BAS

Your BAS environment is set up and managed using the objects described in Table 3.

Table 3. Resource administration objects

| Object name | Description | Function |
|-------------|--|---|
| RASGNDEF | Resource assignments | <p>A resource assignment describes the characteristics of the selected resource definition type, and how those resources are to be assigned to CICS systems. The assigned resources must all be of one type, for example, files, and must belong to a resource group. A resource can be assigned as both local and remote in one or more CICS systems. Before you can use the resource assignment, you must associate with a resource description; see the REDESC object.</p> <p>This object is also used to identify the attributes to be used in selecting the resources to be assigned and any attribute values to be changed when the resources are assigned.</p> |
| RASINDSC | Resource assignments in resource description | <p>This view displays information about resource descriptions, and the resource assignments associated with them. This information includes, for each resource, its resource group and any assigned CICS system to CICS system group.</p> |
| RASPROC | Resource assignment process | <p>This view displays the resources that will be processed when the specified resource assignment is processed. The resources displayed are selected from those in the associated resource group, using any supplied selection criteria.</p> |

Table 3. Resource administration objects (continued)

| Object name | Description | Function |
|-------------|--|--|
| RDSCPROC | Resource description process | This view displays the resources that will be selected when a specified resource description is processed. The resources can be selected from resource groups that are directly related to the resource description, as in migration form BAS, and from resource assignments, using any selection criteria currently in effect. |
| REDESC | Resource descriptions | A resource description identifies a set of resource groups, and hence resource definitions. You use this object to specify whether or not you want to use logical scoping for this resource description, and the CICS systems for the associated resource groups. |
| RESGROUP | Resource groups | This object is used to associate one or more related resource definitions. The resource definitions may be of the same or different types. |
| RESINDSC | Resource groups in description | This view displays information about existing resource descriptions and the resource groups associated with them. |
| RESINGRP | Resource definitions in resource group | This view displays information about existing resource groups and the resource definitions associated with them. |
| SYSLINK | System links | This view displays information about the links that exist between CICS systems in the CICSplex. The information includes the names of the CICS systems and the names of the connection and session definitions used in defining the link. |
| SYSRES | CICS system resources | This view displays the resources that will be assigned to a specified CICS system. The resources are selected on the basis of the resource descriptions currently associated with the CICS system. |
| resDEF | Resource definition | For each resource definition type, there is a resource definition object that defines the attributes for that definition. The available resource types for CICSplex SM are: CONNDEF, DB2CDEF, DB2EDEF, DB2TDEF, DOCDEF, ENQMDEF, FENODDEF, FEPODEF, FEPRODEF, FETRGDEF, FILEDEF, FSEGDEF, JRNLDEF, JRNMDEF, LSRDEF, MAPDEF, PARTDEF, PROCDEF, PROFDEF, PROGDEF, PRTNDEF, RQMDEF, SESSDEF, TCPDEF, TDQDEF, TERMDEF, TRANDEF, TRNCLDEF, TSMDEF, TYPTMDEF |

The object models showing the relationships between these objects are shown in Figure 14 on page 47 and Figure 15 on page 48.

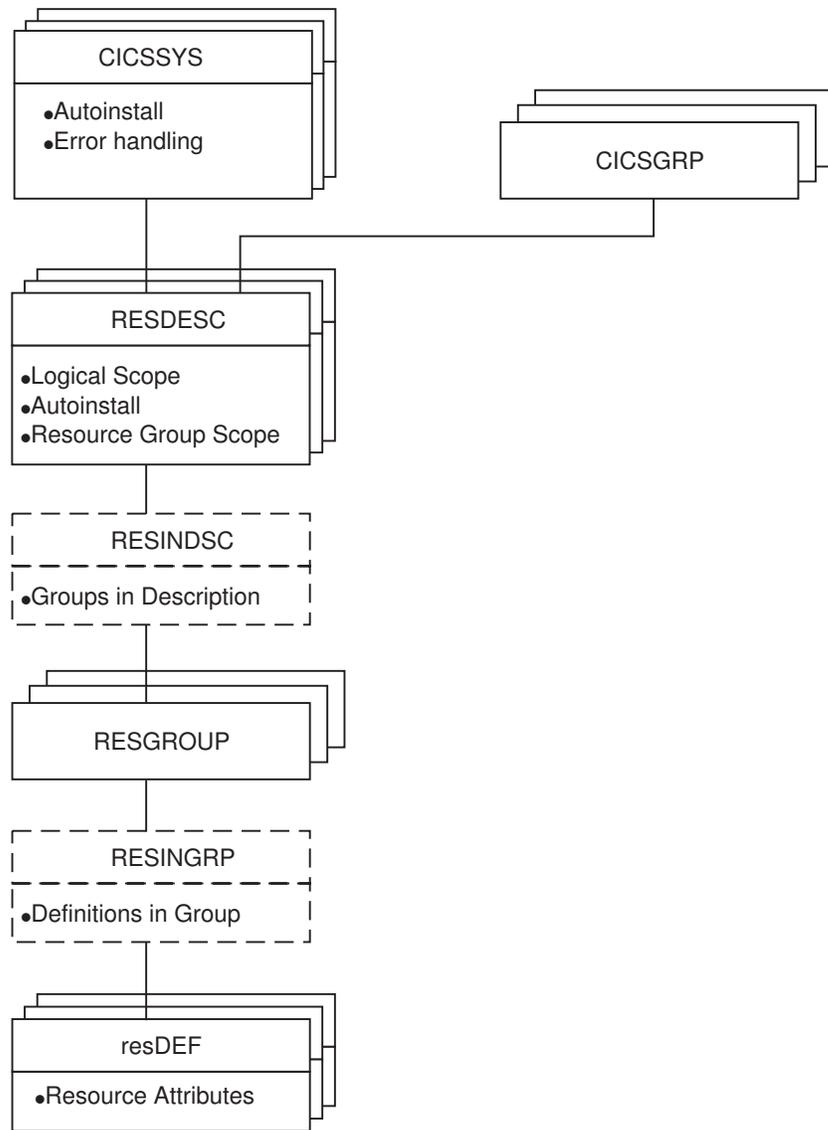


Figure 14. The migration from BAS object model

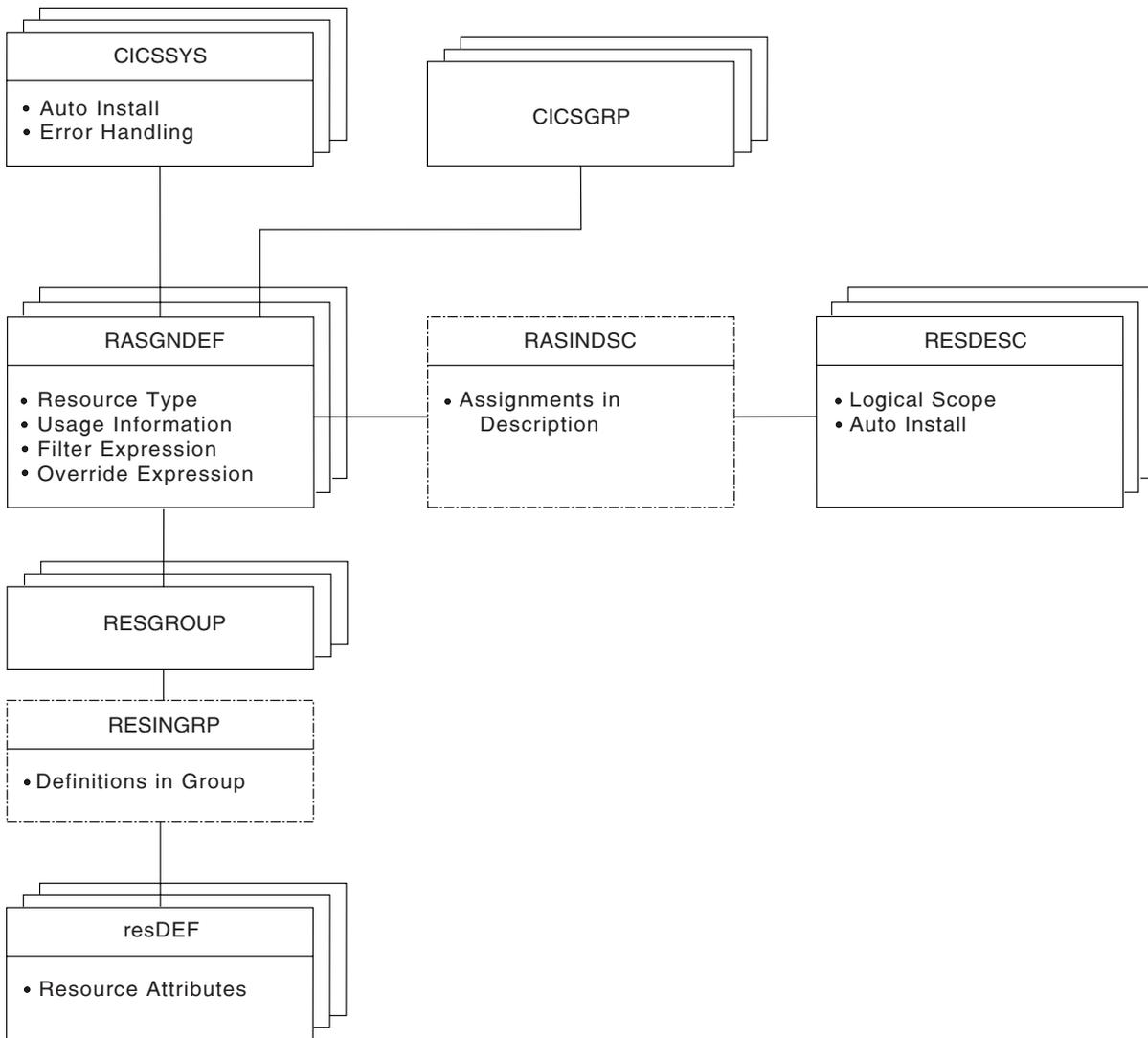


Figure 15. The full-function BAS object model

Migrating your resource definitions

You do not have to recreate your resource definitions and resource groups in BAS. You can migrate your existing CSD structures.

The process of moving your resource definitions and resource groups from your CSDs to the data repository is:

1. Use the EXTRACT command of the DFHCSDUP utility routine to read the CSD records.
2. Use the CICSPlex SM extract routine EYU9BCSD to create batched repository-update facility commands.
3. Input these commands to the batched repository-update facility to create resource definitions and resource groups, and the objects that link them. See *CICSPlex System Manager Administration* for more information of the batched repository-update facility.

This process maintains the relationship between resources and groups. You can migrate all or part of a CSD, or more than one CSD, at any one time. Further information on how to extract CSD records is given in *CICSplex System Manager Managing Business Applications*. Details on using the batched repository-update facility are in *CICSplex System Manager Administration*.

Defining resources

BAS enables you to define and maintain your resource definitions by providing a view for each resource type. See Table 3 on page 45.

The definition of resources for CICSplex SM is similar to CICS RDO. To define a resource, you create a *resource definition object*. You describe the attributes of the resource in the definition but you do not need to define every occurrence of every resource; you can use a small number of resource definitions as templates for the creation of large numbers of resource definitions. The resource definitions are stored in the data repository for the CICSplex.

The differences between CICSplex SM resource definition objects and CICS resource definitions are:

- The same resource definition can be used by any CICS system in the CICSplex, across all supported platforms.
- You can define to CICSplex SM all the attributes of each resource, including both local and remote values. CICSplex SM determines the correct subset of attributes to use when the resource definition is assigned to a CICS system.
- You do not need to define every instance of your resources from scratch. You can create a resource definition as a “template” for many resources with similar, even identical, attribute values. You can specify temporary or permanent variations, called *overrides*, for CICSplex SM to create resources with different sets of values.
- You can create multiple versions of the same named resource definition. Each version is effectively a different resource definition, and can be used by different CICS systems or for different system requirements. For example, you might have different resource requirements for development and test systems. See “Validating resources.”

You can create resource definitions in three ways:

- Using the batched repository-update facility (see “The batched repository-update facility” on page 15)
- Using the BAS administration and definition views
- Using the API; see *CICSplex System Manager Application Programming Guide*.

Validating resources

BAS performs many of the same resource definitions checks as RDO does. As individual resources are defined and installed, BAS checks:

- Individual attributes of a resource
- Interdependent resources attributes
- Release-specific resource attributes

In addition, BAS checks that each *set* of resources is consistent. Every time you make a change to a resource set, BAS checks that the resource being added or updated is not in conflict with a resource already in the CICS system. For example,

you would get an inconsistent set error if you tried to assign different versions of the same resource to the same CICS system, or assign a resource as both local and remote.

BAS also provides a MAP function that allows you to check that your resource associations are what you want. You use this facility to display the structure of your resource associations, starting at any point in hierarchy. Figure 16 shows an example of a resource definition map.

```

----- MAP of RESDESC EYUBAA01 ----- Row 1 of 4
COMMAND ==>                               Scroll ==> PAGE
RESDESC   RASINDSC   RASGNDEF   RESGROUP   RESTYPE   RESDEF
-----
EYUBAD01-+-<----->---EYUBAA01---EYUBAG01-+-CONNDEF -+-C001      0
          |
          +------>---EYUBAG02-+-SESSDEF  ---S001
                                +-SESSDEF  ---S002

```

Figure 16. Example of a BAS resource map

Defining resource groups

A *resource group* is a set of related resource definitions that you want to manage as a unit. Resources groups are defined using the RESGROUP object. The resource definitions in a group may be of the same type or of different types, but usually have something in common. They might be logically related by their use in a given application or communications network, or geographically related by their use at a given site.

There is no real limit to the number or combination of resource definitions that can make up a group. However, only one version of any given resource can be included in a resource group at one time. You have to maintain each version of a resource definition in a different resource group.

Using migration form BAS

With migration form BAS, your resource groups are associated directly with *resource descriptions*. A resource description specifies whether or not you are using logical scoping, and the CICS systems for the resource groups associated with the resource description. See Figure 14 on page 47.

When you have migrated or defined your resources and resource groups, you need to create resource descriptions to define your applications. Resource descriptions are defined using the RESDESC object. A resource description is directly related to one or more resource groups that are to be managed as an entity. In the migration form BAS model, this relationship can be considered to represent a logical scope, but the resource description is really analogous to a RDO group list, in that all the resources are tied to a given CICS system or CICS system group.

Using full-function form BAS

With full-function BAS, resource groups are not associated directly with resource descriptions. An additional object, the *resource assignment* (RASGNDEF), defines the characteristics and usage of a selected resource type from a resource group. The resource assignment is associated with a resource description, which can then be used to represent a business application. See Figure 15 on page 48

These objects are used to define your resources in terms of their business application.

Using resource assignments

Power and flexibility in the management of resources is offered by the *resource assignment* (RASGNDEF) object.

Note: If you are using the migration form of BAS, you do not use resource assignments.

Each resource assignment relates to one resource type within a resource group, and it must be associated with a resource description, if it is to be used in logical scoping or automatically installed. Each resource group can be in more than one resource assignment.

You can further refine the selection of resources within the type specified by the resource assignment by using a filter expression, and you can change the resource attributes using an override expression. You can combine in the expression an unlimited number of attribute values, using the logical operators AND, OR, and NOT. When the resource description is specified as the scope, CICSplex SM processes only those resources that meet the specified selection criteria. This gives you a great deal of control over the management of your resources.

Each resource assignment must be added to a resource description, if it is to be installed and used in logical scoping. Creating resource assignments and adding them to a resource description enables you to manage sets of resources with a logical scope that can span many CICS systems. In this case, the resource description becomes, in effect, a user-defined, logical set of resources, such as an application. You might have several different resource descriptions associated with a given CICS system, each one representing a different set of resources.

For example, with a resource assignment, you can:

- Select specific resources from a resource group.
- Identify the CICS systems where local and remote instances of a resource should be assigned.
- Temporarily override the values of specific resource attributes.

The resources selected by a resource assignment cannot be managed independently. The resources must be members of a *resource group* and the resource assignment must be associated with at least one resource description.

Using resource descriptions

With full-function BAS, a resource description represents an application, that is, a set of logically-related resources that can span more than one CICS system. The resource description thus identifies the *logical scope* of the application.

Each resource group to be used as part of the application must be added to the resource description. You can associate whole resource groups with a resource description to create a larger set of resources (much like a CSD group list). This allows you to manage the resources more efficiently. The set of resources identified in a resource description can be:

- Identified as a logical scope (such as an application) for use in subsequent CICSplex SM requests

- Installed either automatically, when the CICS system identifies itself to CICSplex SM as a MAS, or dynamically, while the CICS system is up and running.

The BAS objects that are used for creating and managing these resource associations shown in Figure 14 on page 47 and their functions are summarized in Table 3 on page 45.

For more information about resource descriptions see the *CICSplex System Manager Managing Business Applications*.

Installing applications and resources

Optionally, as an alternative to installing resources from the CSD, you can use BAS to install your resources, from the data repository. BAS allows you to install all the resources associated with an application, by installing the resource description, or one resource group. BAS applies any resource assignments you have associated with the application resource description or resource group, and selects the actual resources to be installed on the basis of those criteria. Alternatively, you can install an individual resource into in one or more CICS systems, either locally or remotely, as appropriate.

BAS can install resources either automatically at system initialization time or dynamically into an active CICS system. Before installing a resource, CICSplex SM performs checks to determine whether the resource already exists in the CICS system. The normal situation is that if the resource exists, the new resource is considered a duplicate and is not installed. However, when you dynamically install resources, BAS offers the option of bypassing the duplicate resource checking and forcing the unconditional installation of resources.

Automatic installation

When a CICS system initializes and identifies itself to a CMAS, BAS reviews the resources associated with that system and decides the set of resources to be installed.

Dynamic installation

You can install resources dynamically into an active CICS system by issuing the INS action command either from the appropriate resource definition view, or from the Resource group (RESGROUP), Resource description (RESDESC), or System link (SYSLINK) view.

Security considerations

Because of the importance of resource definitions to your CICSplex SM environment, you need to consider carefully the implications of allowing users access to certain types of resource or to certain functions. You can allow a user to have access to all resource definition views and their related administration views, or you can restrict access to the definition views for a particular resource type. You can also define the type of access a user has, for example, one user might have read-only access to connection definition (CONNDEF) and session definition (SESSDEF) views, but update access (which permits creation and administration of resources) to all other resource types. You need to make sure that the BAS views are adequately protected, so that unauthorized users cannot create and administer resources.

You should also take care if you are running CICS TS, and are using the EXEC CICS CREATE command to build new resources. Any definition created with the

CICSplex as the context is automatically distributed to all CMASs in the CICSplex. Therefore, giving a user authority to create BAS objects is potentially equivalent to giving authority to install resources on any CICS system in the CICSplex. When the CICS system starts, there is no check on who installed the resource in the system.

For details on setting up security for CICSplex SM at your enterprise, see *CICSplex System Manager Managing Business Applications*.

Planning for BAS

BAS allows you to create CICSplex SM objects that define your resources and to group them in such a way that you have close control over the running of your applications. You do not have to transfer all your CSD resources to CICSplex SM at the same time, neither do you have to decide at the outset which approach you will use. You can extract the resource definition records from one or more CSDs, or only a subset of records from one CSD. You can move directly to using full-function BAS, or you can use migration from BAS, and gradually move towards full-function BAS. You can use RDO, migration from BAS, and full-function BAS in the same CICSplex.

This section poses some of the questions you might like to consider when you start planning to implement BAS in your enterprise:

- If you are going to use the extract facilities:
 - In what order do you want to extract your CSDs?
 - Do you want all the records from a CSD or a subset?
 - Are you going to extract more than one CSD at a time?
- Are you going to use the migration from approach?
- Are you going to implement full-function BAS? You then need to think about how your business applications use resources and the assignments you need to create.
- Which resources are to be installed automatically and which are to be installed dynamically?
- Where are the resources to be installed?
- Do you need to define manually any resources that are not defined in the CSD?
- What security measures do you need to implement? (See “Security considerations” on page 52).

The IBM CICS Interdependency Analyzer for z/OS can assist you in the planning and understanding of your application resource flow; it shows you:

- Transaction resource dependencies; that is, the sets of resources used by individual CICS transactions, and on which the transactions depend in order to run successfully
- Transaction affinities; that is, those groups of transactions that have affinities with each other, meaning that the whole group must be installed in the same region, or in a particular region

For more information about the CICS Interdependency Analyzer, see the *CICS Interdependency Analyzer for z/OS User's Guide and Reference*.

Implementing BAS

This section describes a route you could take to implement BAS on your CICSplex.

Creating migration form BAS

The first stage in the process of moving to using BAS is to migrate your resources from CICS to CICSplex SM:

- Move your resource definitions, resource groups, and the relationships between them, from the CSD to the CICSplex SM data repository. This process, which is described in “Migrating your resource definitions” on page 48, gives you an intermediary resource hierarchy that is very like CEDA, in that each resource definition is in a resource group.
- Create definitions and groups for any resources that are not held on the CSD.
- Create applications by associating each resource group with a resource description.

There are no resource assignments and, although you can now run your CICSplex SM system perfectly satisfactorily, you will not be taking advantage of the special functionality offered by BAS.

Once you have extracted your CEDA definitions, you can proceed to migrate your resource definitions to full BAS function, as described in “Migrating to the use of resource assignments.”

Migrating to the use of resource assignments

You don't have to migrate all your resources to resource assignments at the same time. You can identify certain resources in which you are interested, remove them from direct association with resource definitions, and create resource assignments (using the RASGNDEF object) for them. When you are happy with the resource assignments, you can move on to defining another set of resource definitions.

As you move towards processing your resource definitions more full-functionally, you should remove resource groups from direct association with a resource description and identify them instead in one or more resource assignments.

Note: You can use the same resource description to manage both whole resource groups and selected resources identified in resource assignments. As you begin to take advantage of the resource assignment capabilities, you may have to update some of the underlying resource definitions. For example, a resource definition that was previously associated as is with a CICS system might require additional attributes before it can be assigned as both a local and remote resource in different CICS systems.

A recommended approach

When you decide to implement BAS functions in your enterprise, you should begin by defining the appropriate objects. This section summarizes the preceding sections. You should:

- Extract resource definitions and resource groups from the CSD, as described in “Migrating your resource definitions” on page 48. Alternatively:
 - Use the appropriate resource definition views to create resource definitions.
 - Use the RESGROUP view to create resource groups.
- Use the RASGNDEF view to create resource assignments.
- Use the RESDESC view to create resource descriptions.

Then, to create associations between these objects:

- Use the appropriate resource definition view to add resource definitions to the resource group.

- Use the RESGROUP view to add the resource group to the resource assignments.
- Use the RASGNDEF view to associate the resource assignment with a CICS system or CICS system groups.

You can use the MAP action command to display a visual map of the BAS definitions in your data repository. If you want to see a list of the objects and associations you have already defined, use the RESINGRP view.

CICSplex SM includes a *Starter Set* of CICSplex SM object definitions that illustrates the concepts and components introduced in this chapter. The Starter Set is described in detail in Chapter 10, “The CICSplex SM Starter Set,” on page 101.

If you have problems defining any particular object or association, refer to the CICSplex SM Starter Set examples. The Starter Set includes BAS examples that you can use as models for your own objects. You can copy the supplied samples directly into your data repository via the batched repository-update facility. For more information about using the Starter Set as a model see the *CICS Transaction Server for z/OS Installation Guide*. For more information about the batched repository-update facility, see *CICSplex System Manager Administration*. The Starter Set definitions are listed in Chapter 10, “The CICSplex SM Starter Set,” on page 101.

Where next?

Once you have identified your BAS requirements, you need to set up your resources and their associations as described in *CICSplex System Manager Managing Business Applications*.

a recommended approach

Chapter 6. Managing workloads

This chapter covers:

- “What is workload management?”
- “Workload balancing” on page 61
- “Workload separation” on page 63
- “Intertransaction affinity” on page 64
- “Managing workloads” on page 65
- “Planning for WLM” on page 67
- “Implementing WLM” on page 69
- “Where next?” on page 72

For full details, see *CICSplex System Manager Managing Workloads*.

What is workload management?

CICSplex SM's workload management (WLM) function provides you with the capability to control where your work requests are run, in order to optimize performance and workload throughput. WLM uses the CICSplex SM dynamic routing program EYU9XLOP to route work requests to the most suitable target region from a predefined set of target regions.

What is dynamic routing?

In a CICSplex or BTS-set, resources such as transactions and programs required in one region may be owned by another. For example, you may have a terminal-owning region (TOR) that requires access to transactions owned by an application-owning region (AOR).

You can specify the location of a resource when you are designing your system. Then, requests of a specific resource are always routed to the same region. Typically, the location of the resource is specified in the installed resource definition. This is known as *static routing*.

With *dynamic routing*, the location of the resource is decided at run time. In CICSplex SM, the decision on where to run a piece of work is made by the user-replaceable dynamic routing program (called the dynamic *transaction* routing program in previous releases). The user-replaceable program EYU9XLOP creates the environment necessary for CICSplex SM-based dynamic routing, and sets up the run-time environment.

The dynamic routing program can route:

- Transactions initiated at a terminal
- Eligible EXEC CICS START requests that are associated with a terminal
- Eligible EXEC CICS START requests that are not associated with a terminal
- Dynamic program link (DPL) requests that are received using:
 - The CICS Web Interface
 - The CICS Gateway for Java
 - External CICS interface (EXCI) client programs
 - Any CICS client workstation products using the External Call Interface (ECI)
 - Distributed Computing Environment (DCE) remote procedure calls (RPCs)
 - Open Network Computing (ONC) RPCs
 - Internet Inter-Object Request Block Protocol (IIOP)

- The Link3270 bridge
- Any function that issues an EXEC CICS LINK PROGRAM request
- CICS business transaction services (BTS) processes and activities
- Enterprise beans executing in CICS-provided CorbaServers

In CICSplex SM, dynamic routing is managed by the Workload Manager component of CICSplex SM.

The CICS regions involved in dynamic routing may act as one or more of the following:

Requesting region

The CICS region in which the work request originates.

Routing region

The CICS region in which the decision is taken on where the work will run.

Target region

The CICS region where the request is actioned.

For *dynamic transaction routing*, the requesting region and the routing region are typically TORs, and the target region is typically an AOR.

For *inbound DPL client requests*, the requesting region and the routing region are typically TORs, and the target region is typically an AOR.

For *EXEC CICS START commands associated with a terminal*, the requesting region is typically an AOR, the routing region is typically a TOR, and the target region is typically an AOR.

For *peer-to-peer DPL requests, EXEC CICS START commands that are not associated with a terminal, for CICS business transaction services processes and activities, and for Link3270 bridge requests*, the requesting region, routing region, and target region are typically AORs.

For *enterprise bean invocations*, the requesting region is typically the external client code (but can be Enterprise JavaBean code in another CICS region) that invokes the enterprise bean, the routing region is a CICS listener region, and the target region is typically an AOR.

Dynamic routing models

There are two possible dynamic routing models:

- The traditional “hub” model
- The distributed model

The traditional hub model

The traditional hub model is shown in Figure 17 on page 59. This is the model used for the dynamic routing of transactions, EXEC CICS START commands associated with a terminal, and inbound client DPL requests. The request is initiated in the requesting region, typically a TOR, which also acts as the routing region. The request is routed to a target region, selected from the specified target group, where the program is executed.

The “hub” model is *hierarchical*, in which routing is controlled by one region (the routing region, that is, the TOR). Normally, a routing program runs only in the routing region. This model has the advantage of being relatively simple to

implement. For example, compared with the distributed model, there are few inter-region connections to maintain. The disadvantages of the hub model are:

- If you use only one hub to route transactions and program-link requests across your target regions, the hub routing region is a single point-of-failure.
- If you use more than one hub to route transactions and program-link requests across the same set of target regions, you may have problems with distributed data. For example, if the routing program keeps a count of routed transactions for load balancing purposes, each hub routing region will need access to this data, which may be maintained in a local temporary storage queue.

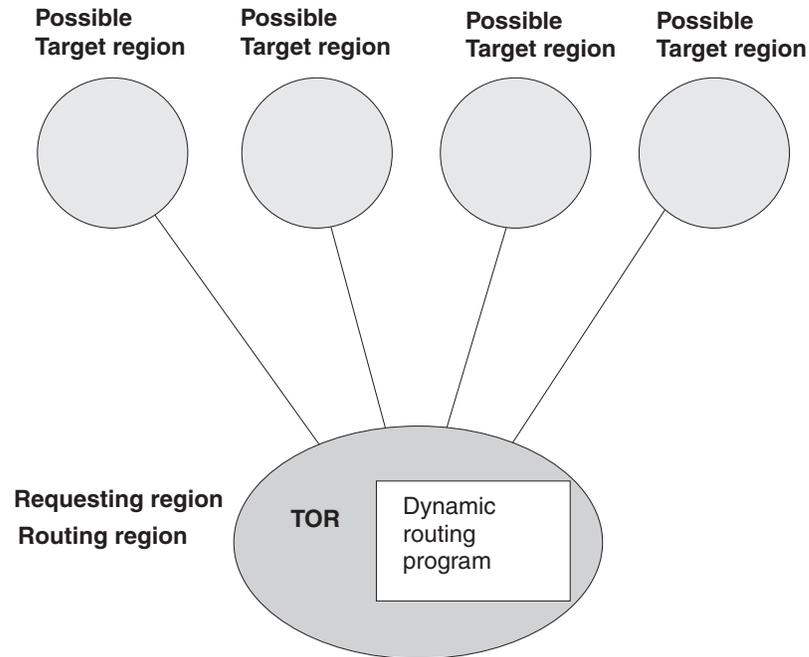


Figure 17. Dynamic routing using a hub routing model

The distributed model

The distributed model is shown in Figure 18 on page 60. This is the model used for the dynamic routing of EXEC CICS START requests that are not related to a terminal, enterprise bean and BTS activities, and Link3270 bridge requests. Each CICS system in the target group may act as a requesting region, routing region, and target region. A distributed routing program runs in each region.

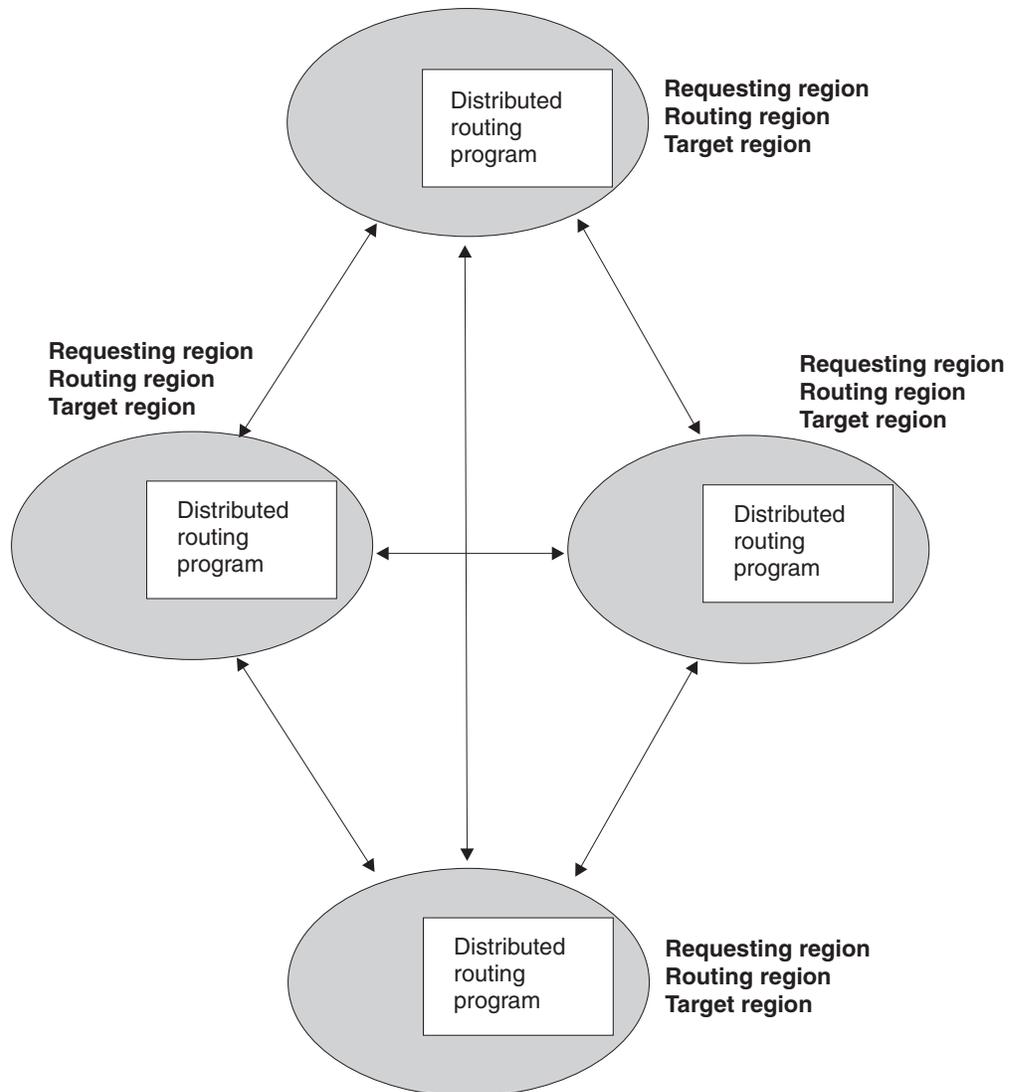


Figure 18. Dynamic routing using a distributed routing model. Note that, for CICSplex SM, the dynamic routing program EYU9XLOP performs also the distributed routing function.

The advantage of the distributed model is that there is no single point of failure. The disadvantages are:

- Compared with the hub model, there are a great many inter-region connections to maintain.
- You may have problems with distributed data. For example, any data used to make routing decisions must be available to all regions. With CICSplex SM, this problem is solved by the use of data spaces.

WLM functions

CICSplex SM's dynamic routing program supports:

- *Workload separation*: see page 63
- *Workload balancing*: see page 61
- *Intertransaction affinity*: see page 64

Advantages of WLM

CICSplex SM's WLM function is of particular benefit in those enterprises that are running CICS/ESA on Parallel Transaction Servers (PTSs), because CICSplex SM can route work throughout the sysplex.

With WLM in your enterprise, you have:

- The ability to route all types of program link request dynamically to improve the performance and reliability of inbound client and peer-to-peer DPLs.
- The ability to route EXEC CICS START TRANSID TERMID commands dynamically to improve the performance and reliability of the applications using these commands.
- The ability to integrate workload balancing for terminal-initiated transactions, non-terminal-initiated transactions, EXCI clients, CICS clients, CICS Web support, CICS Transaction Gateway, IIOP, and started tasks.
- The ability to integrate BTS processes and activities fully into the workload separation and workload balancing functions.
- The ability to integrate enterprise bean invocations into the workload balancing and workload separation functions.
- The ability to perform workload balancing and separation for Link3270 bridge requests.
- Optimum performance and response times for a variable and unpredictable workload.
- Work routed away from a failing target region to an active target region.
- Opportunities for increased throughput and improved performance.
- Reduced risk of bottlenecks
- Individual target regions taken out of service without impact to the end-user.
- Less operator intervention.

For full details, see *CICSplex System Manager Managing Workloads*.

Workload balancing

Workload balancing is the routing of transactions or programs among a group of target regions according to the availability and activity levels of those target regions. Workload balancing can be used in addition to, or in place of, workload separation. For example, CICSplex SM can balance the workload among the SALESGRP target regions by selecting, as each transaction is initiated, the target region that is likely to deliver the best performance.

Which target region processes the work is determined wholly by CICSplex SM using one of two algorithms. These are the *queue algorithm* and the *goal algorithm*.

Workload balancing is statistical in nature. Selection of the appropriate target region is based completely on the target region's ability to achieve the expected response time when utilizing the goal algorithm, or balancing the load across a set of target regions when using queue. If all the target regions in a set are equally capable of handling the work within the constraints of link type, abend probability, health, normalized load and response time (for goal), then a target region is randomly selected from the resulting set. Therefore, in systems that are lightly loaded, there is no predetermined order in which work is allocated to equally capable target regions, since the target regions are by definition equally capable of achieving the desired effect. This is in contrast to some balancing algorithms that use a "round

robin” technique, whereby work is allocated to the given set of target regions simply by allocating the next instance to the next target region in the ring.

When selecting a target region, both the queue algorithm and the goal algorithm take into account the way in which a target region is connected to its requesting region. That is, a target region connected to its requesting region via MRO/XCF is preferred to a VTAM[®]-connected target region, when all other considerations are equal. For workload balancing of enterprise-bean related requests, only MRO connections between routing regions and target regions are supported.

Workload balancing of enterprise beans can be achieved using the queue and goal algorithms. The inbound IIOP work request is received by a routing region (listener) and is matched to a bean name, an operation and a CorbaServer using a request model definition. The routing region routes the transaction identified in the request model to a target region. The transaction runs in the CorbaServer corresponding to the installed request model instance.

The queue algorithm

The queue algorithm causes CICSplex SM to select the target region that:

- Has the shortest queue of work waiting to be processed, relative to the maximum number of tasks permitted in the target region. This queue of work, also called the load count, is the count of all active and queued user tasks. By default, tasks queued for both MAXTASKS and TRANCLASS are included in the load count. Use of the WLMLOADCOUNT EYUPARM allows sites to exclude tasks queued for TRANCLASS from the load count. Please refer to the *CICS Transaction Server for z/OS Installation Guide* (GC34-6426) for more information about the WLMLOADCOUNT EYUPARM.
- Is the least affected by conditions such as short-on-storage, SYSDUMP, and TRANDUMP.
- Is the least likely to cause the transaction to abend. This factor may be taken into account for CICS Transaction Server for OS/390, Version 1 Release 3 (and later), target regions only.

This algorithm maximizes work throughput and standardizes response times across the CICSplex. The queue algorithm is very robust: it can accommodate differences in processor power; different maximum task values in the target regions; asymmetric target region configurations; and an unpredictable workload.

The goal algorithm

The goal algorithm causes CICSplex SM to select the target region that:

- Is the least affected by conditions such as short-on-storage, SYSDUMP, and TRANDUMP.
- Is the least likely to cause the transaction to abend.
- Is the most likely to enable the transaction to meet average response-time goals set for it (and for other transactions in its class) using the Workload Manager component of MVS/ESA 5.1 (or later).

MVS/ESA 5.1 (or later) is a prerequisite of the goal algorithm. Also, routing regions in a CICSplex using the goal algorithm must be CICS TS regions.

The goal algorithm works best in environments where, if the CICSplex crosses multiple MVS images:

- A similar number of target regions is on each of those MVS images.

- The number of service classes on each MVS image is similar to the number of target regions on each MVS image.

Cross CEC routing

CPSM routing regions use task load percentages ((active tasks/maxtask) *100) in
 # available target regions as part of the determination for which target region to route
 # to. The percentage value, which is set by the WLMLOADTHRS EYUPARM, must
 # be met by all target regions on the same CEC as the routing region before the
 # routing region routes to target regions on other CECs, if all other health factors (for
 # example, short-on-storage) for the target regions are similar. Once all target regions
 # on other CECs meet this value, the routing region resumes routing to the target
 # regions on the local CEC again. When a local target's task load drops below this
 # value the routing region resumes routing to that target region regardless of the task
 # load in target regions on remote CECs.

For example, if the default value of 65 is used, then all target regions on the CEC
 # where this routing region resides must have a task load of 65% or higher before the
 # routing region routes to target regions on other CECs, if all other health factors (for
 # example, short-on-storage) for the routing regions are similar. As soon as target
 # regions on other CECs achieve a task load of 65% or higher, the routing region
 # resumes routing to the target regions on the local CEC again. When a local target
 # region's task load drops below 65%, the routing region resumes routing to that
 # target region regardless of the task load in target regions on remote CECs.

Specifying this value lower than the default probably decreases the delay in routing
 # to target regions on remote CECs. Take care not to set this value so low that the
 # threshold is met by long-running tasks in the target regions.

Specifying this value higher than the default most likely increases the delay in
 # routing to target regions on remote CECs.

Note that the effectiveness of this parameter is increased as the characteristics (for
 # example, maxtask value or number of long running tasks) of the target regions
 # become similar.

Workload separation

Workload separation routes work from a requesting or routing region among a set of target regions, but the requirements of users, terminals, the transactions themselves, and their BTS process type, can influence which target region set is used. The routing of particular transactions or programs to a particular group of target regions is based on any combination either of user ID, terminal ID, and transaction group. For BTS processes, routing is based on a combination of transaction group and process type. For enterprise bean-related transactions, routing is based on transaction group. For Link3270 bridge requests, separating by terminal luname is subject to some restrictions. See *CICSplex System Manager Managing Workloads* for more information.

For example, you can specify that:

- Any transaction initiated by users whose user IDs begin with the characters "PAY" must be routed to a target region in CICS system group AORPAY1.
- Any transaction initiated from a terminal whose LU name begins with the characters "NYORK" must be routed to a target region in CICS system group AORNYORK.

- Any transaction belonging to the transaction group ACCOUNTS must be routed to a target region in CICS system group AORACCNT.
- All processes associated with a BTS process type TRAVEL are routed to a target region in one BTS-set, and all processes associated with BTS process-type PAYROLL are routed to another BTS-set.
- All enterprise bean-related transactions belonging to transaction group STOCK are routed to any target region in CICS system group STOCKTG, if the user id matches that in the workload definition.

Intertransaction affinity

An intertransaction affinity is a relationship between transactions, of a specified duration, that requires them to be processed by the same target region. For example, you might have a pseudoconversation made up of three separate transactions, and each transaction passes data to the next transaction in the sequence via a temporary storage queue (which may be shared in the distributed model). You would then specify that all three transactions must be processed by the same target region, and that this affinity lasts for the duration of the pseudoconversation. (If you did not define this affinity to CICSplex SM, each transaction could be routed to a different target region and would therefore be unable to access temporary-storage data left by the previous transaction.) The target region itself is selected by CICSplex SM from the specified target scope.

Workload management and the IBM CICS Interdependency Analyzer for z/OS understand affinities between BTS processes and activities. BTS itself does not introduce affinities, and discourages programming techniques that do, but it does handle existing code that may introduce affinities. You should define such affinities to workload management, so that it can make sensible routing decisions. It is particularly important to specify each affinity's lifetime; failure to do so may restrict unnecessarily workload management's routing options.

Workload management and the IBM CICS Interdependency Analyzer for z/OS do not understand affinities between routable non-terminal-related EXEC CICS START commands, or between DPLs not associated with a user id or a terminal. You should take steps either to remove any affinities from your applications, or to ensure that your applications honor any affinities.

Note that, if data is passed between transactions via the COMMAREA on the EXEC CICS RETURN command, no such affinity exists: the COMMAREA is passed back to the requesting region, and so can be passed to the target selected to process the next transaction in the sequence. For more information about ways of avoiding or minimizing transaction affinities, see the CICS/ESA publication *Dynamic Transaction Routing in a CICSplex*.

How intertransaction affinities are honored

When the first transaction from a group of related transactions is started, CICSplex SM selects an appropriate region from the specified target scope. If there is more than one suitable region in the target scope, CICSplex SM selects one using the current workload balancing algorithm. Subsequent transactions in the same group that meet the affinity criteria are directed to the same region as the first transaction. If subsequent transactions do not meet the affinity criteria (for example, if the same pseudoconversation is started from a different user ID), the selection process for a suitable region starts again.

Managing workloads

Your workload environment is set up and managed using the objects described in Table 4.

Table 4. Workload management objects

| Object name | Description | Function |
|-------------|---|---|
| DTRINGRP | Transactions in transaction groups | This object displays information about the transactions associated with one or more transaction groups. |
| TRANGRP | Transaction groups | A transaction group is an association of logically similar transactions. The similarity may be based on affinity requirements, common shared processing requirements, or any other user-determined characteristics. The transaction group is included as part of the workload definition, which in turn defines the CICS system group to be used as the target region. Any transaction that is not specifically associated with a transaction group is assigned to the default transaction group. |
| WLMATAFD | Transaction affinities for an active workload | This object shows a detailed display of the properties of any single active affinity. This is useful, for example, when you need to view the contents of a CICS BTS affinity key in hexadecimal format. |
| WLMATAFF | Transaction affinities for an active workload | This object shows information about the active affinities for a transaction group associated with a workload within the CICSplex identified as the context. An affinity becomes active when the first transaction associated with the transaction group is dynamically routed to a target region. |
| WLMATAFS | Summary of transaction affinities | This object, which is the summary form of the WLMATAFF object, shows summarized information about the active affinities for a transaction group associated with a workload with the CICSplex identified as the context. |
| WLMATGRP | Transaction groups associated with an active workload | This object shows information about transaction groups associated with a workload that is within the CICSplex identified as the context. |
| WLMATRAN | Transactions associated with an active workload | This object shows information about all active transactions associated with a workload that is within the CICSplex identified as the context. |
| WLMAWAOR | Target regions in an active workload | This object shows information about all target regions that are associated with a workload that is within the CICSplex identified as the context. |
| WLMAWAOS | Target regions in an active workload summary | This object shows summarized information about all target regions that are associated with a workload that is within the CICSplex identified as the context. |
| WLMAWDEF | Active workload definitions | This object shows information about active workload definitions associated with a workload that is within the CICSplex identified as the context. |

Table 4. Workload management objects (continued)

| Object name | Description | Function |
|-------------|--|---|
| WLMWORD | Details of an active workload | This object shows information about an active workload within the CICSplex identified as the context. A workload is active within a CICSplex as long as: <ul style="list-style-type: none"> • A CICS system that is acting as a routing region or target region, and is participating in the workload, is connected to that CICSplex. • Any transaction causes an affinity lifetime of PERMANENT to be established. |
| WLMWORK | Active workloads | This object shows information about active workloads within the CICSplex identified as the context. A workload is active within a CICSplex as long as: <ul style="list-style-type: none"> • A CICS system that is acting as a routing region or target region, and is participating in the workload, is connected to that CICSplex. • Any transaction causes an affinity lifetime of PERMANENT to be established. |
| WLMAWTOR | Routing regions in an active workload | This object shows information about routing regions that are associated with a workload that is within the CICSplex identified as the context. |
| WLMAWTOS | Routing regions in an active workload | This object shows summarized information about routing regions that are associated with a workload that is within the CICSplex identified as the context. |
| WLMDEF | Workload definitions | This object identifies which transactions are to be routed to which CICS system group. You identify the transactions by transaction group, and within the group either by LU name and user id, or by process type. |
| WLMGROUP | Workload groups | This object is used to associate one or more workload definitions. |
| WLMINGRP | Workload definitions in workload groups | This object displays the names of workload groups and the workload definitions associated with them. |
| WLMINSPC | Workload groups in workload specifications | This object displays the names of workload specifications and the workload groups associated with them. |
| WLMSCOPE | Workload specifications assigned a scope | This object displays information about the CICS systems or CICS system groups acting as routing regions that are associated with a named workload specification. |
| WLMSPEC | Workload specifications | A workload specification identifies a workload and one or more CICS system groups acting as target regions. It also defines the attributes of the default transaction group. |

The WLM object model is shown in Figure 19 on page 67.

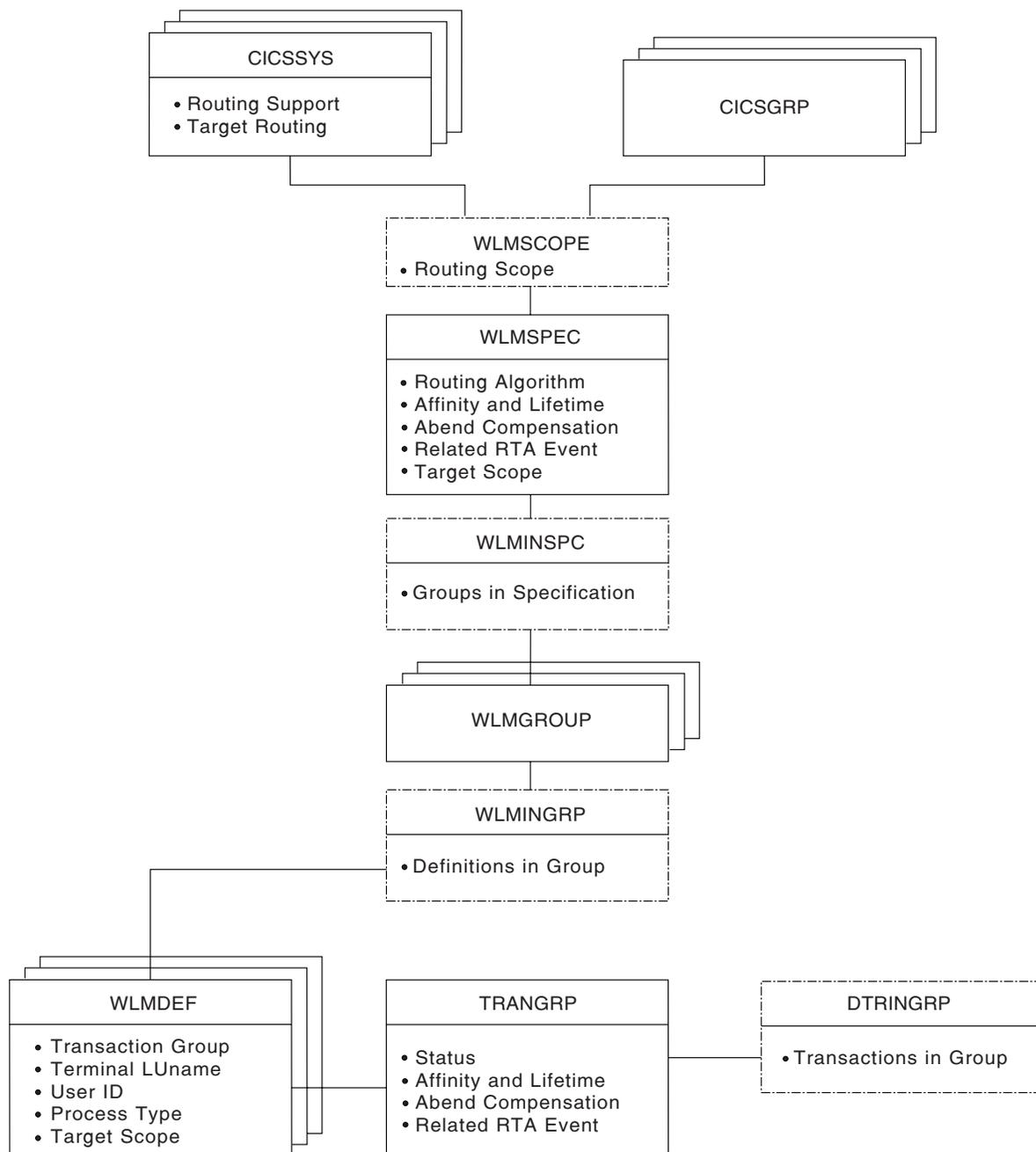


Figure 19. The WLM object model

Planning for WLM

This section provides some instructions to help you determine the extent to which you can use CICSplex SM's workload management in your enterprise.

Workload balancing or workload separation?

Workload balancing should be used wherever possible because it makes the best use of the available CICS systems, and provides opportunities for increased throughput and performance.

Workload separation (by process type, user, terminal, and transaction) should be implemented only where strictly necessary, because it prevents full exploitation of CICSplex SM's workload balancing functions. If you have defined more than one CICSplex, and have made the division to reflect use of CICS systems by different groups of users, for example, it's possible that you will be able to use simple workload balancing rather than workload separation within the CICSplex. For example, if you have separated the CICS systems used by group A from the CICS systems used by group B by defining two CICSplexes, then within each CICSplex you can implement workload balancing. If you haven't taken this approach, you might consider it necessary to recognize such groupings by implementing workload separation.

There are two major activities in your planning for WLM:

- Identifying the workloads in your enterprise
- Identifying intertransaction affinities and trying to remove them

Identifying the workloads

Begin by identifying the workloads processed in your enterprise. These are certain to be apparent in any underlying TOR–AOR–FOR configurations that existed prior to your interest in CICSplex SM. Next, confirm that the current CICSplex SM configuration of CICS systems supports the identified workloads. In particular:

- Routing regions and target regions from a single workload must be in the same CICSplex. That is, the supplied dynamic routing program cannot route transactions beyond the confines of the CICSplex. (It is possible to route transactions outside of the CICSplex by customizing the supplied dynamic routing program. Customization of the supplied dynamic routing program is described in *CICSplex System Manager Managing Workloads*.)
- A routing region must be:
 - A CICS TS region.
 - A local MAS, that is, the routing region cannot be running on an MVS image on which there is no CMAS.
 - In only one workload, that is, the routing region can be associated with only one active workload specification at a time.
 - For a BTS transaction, a terminal-related EXEC CICS START command, a non-terminal-related EXEC CICS START command, and a dynamic program link, CICS Transaction Server for OS/390 Version 1 Release 3 and later.
 - For an enterprise bean invocation, CICS Transaction Server for z/OS, Version 2 Release 2 and later.
 - For a Link3270 bridge request CICS Transaction Server for z/OS, Version 2 Release 3 and later.
- A target region can be:
 - a local MAS
 - in multiple workloads
 - any CICS system managed by CICSplex SM
 - For a BTS transaction, a terminal-related EXEC CICS START command, a non-terminal-related EXEC CICS START command, CICS Transaction Server for OS/390 Version 1 Release 3 and later
 - For an enterprise bean invocation, CICS Transaction Server for z/OS, Version 2 Release 2 and later.
 - For a Link3270 bridge request CICS Transaction Server for z/OS, Version 2 Release 2 and later.

Identifying intertransaction affinities

Intertransaction affinities, which require related transactions to be processed by the same target region, prevent optimum workload distribution. In general, they arise either because of the way in which one transaction passes data to another, or because of a requirement to coordinate the processing of two or more transactions. Identifying affinities isn't always easy, but there are some methods you can use. For example, you can review application design documentation or source code; you can run CICStraces; or you can use the IBM CICS Interdependency Analyzer for z/OS. For a detailed discussion of intertransaction affinities and ways of identifying them, see the CICS/ESA publication *Dynamic Transaction Routing in a CICSplex*.

Having identified any affinities in a workload, you should make every attempt to remove them. If you cannot remove them completely, gauge the duration of the affinity and try to minimize it. You can define an intertransaction affinity to CICSplex SM as lasting:

- While the user's session is active
- For the duration of the terminal session
- While the target region remains active
- While the workload is active
- For the duration of a pseudoconversation
- While the BTS activity is active
- While the BTS process is active

Be aware that CICSplex SM must honor an active affinity: if an affinity is active but the target region becomes unavailable, the transaction isn't routed. In the case of a BTS transaction, BTS will wait for the region to start.

Implementing WLM

What you have to do to use CICSplex SM's workload management functions varies depending on whether you want to implement workload balancing only, or whether you also want to implement some workload separation, and possibly define one or more transaction affinities in the same CICSplex. You are recommended to follow this process:

1. Identify candidates for dynamic routing
2. Implement workload balancing for each workload in the CICSplex where dynamic routing is possible.
3. Add any necessary workload separation requirements.
4. Define any intertransaction affinities.

Each step is described in the remainder of this section.

Identifying dynamic routing candidates

Not all work requests are candidates for dynamic routing. The conditions that have to apply for a work request to be eligible are listed in the *CICS Intercommunication Guide*.

For further details:

- See the *CICS Application Programming Guide* and the *CICS Application Programming Reference* for descriptions of the EXEC CICS LINK and EXEC CICS START commands.
- See *CICS Business Transaction Services* for a description of the EXEC CICS RUN ASYNCHRONOUS command.

- See either the *CICS Resource Definition Guide* or *CICSplex System Manager Managing Business Applications* for resource definitions.
- See the Information Center section *Java Applications in CICS* for all information on enterprise beans.
- See the *CICS External Interfaces Guide* for more information about the Link3270 bridge.

Implementing workload balancing

To implement workload balancing within a CICSplex, your task is very straightforward:

1. If there is more than one routing region in a single workload, create a CICS system group of those routing regions, using the CICSGRP object.
2. Create a CICS system group of the target regions to which the routing regions can route transactions, using the CICSGRP object. This group could include every CICS system in the CICSplex.
3. Create one workload specification for each workload in the CICSplex, using the WLMSPEC object. In the workload specification, specify the name of the target system group as the target scope value, and identify which algorithm is to be used for workload balancing.
4. Associate the single routing region, or the routing system group, with the workload specification. (Remember that you can have multiple workloads in single CICSplex, but that a routing region or group of routing regions can belong to only one active workload specification at a time.)

Workload balancing is effective within the workload when a routing region belonging to the workload is next started.

Adding workload separation requirements

To implement workload separation in the same workload, by any combination either of user id, terminal id, and transaction group, or of process type and transaction group:

1. Create a CICS system group *for each set of target regions*, using the CICSGRP object. For example, if:
 - Transactions initiated by users ABC* from terminals NETA* must be routed to target regions 2 through 7
 and
 - Transactions initiated by users HIJ* from any terminal must be routed to target regions 5 through 9
 two target system groups must be defined: one group includes target regions 2 through 7, and one group includes target regions 5 through 9.
2. Define a transaction group, using the TRANGRP object, if workload separation by transaction is to be implemented. For example, if transactions EFG1, EFG2, and EFG3 must be routed to a specific group of target regions, you must define a transaction group to include those transactions.
3. Create one workload definition, using the WLMDEF object, for each user, terminal, and transaction group combination. For the example in item 1 (above):
 - One workload definition would be required for the combination of any transaction, users ABC*, and terminals NETA*.
 - One workload definition would be required for the combination of any transaction, users HIJ*, and any terminal.

A third workload definition would be required to support the workload separation by transaction described in item 2 on page 70. In each workload definition, include the name of the appropriate target region system group to which eligible transactions are to be routed.

4. If the workload definitions are to be installed automatically when the first requesting region in the workload starts, you should use the WLMGROUP object to:
 - a. Create a workload group.
 - b. Add the workload definitions to the workload group.
 - c. Associate the workload group with the workload specification.

Automatic installation of the workload definitions is likely to be required, because workload separation is usually a regular and consistent requirement. However, you can also install a workload group or individual workload definitions manually for immediate effect.

Defining intertransaction affinities

If there is a requirement for multiple transactions in the workload to be processed by the *same* target region:

1. Create a transaction group, using the TRANGRP object, for each set of transactions within the workload that share an affinity. For example, if transactions MNO1, MNO2, and MNO3 must be processed by the same target region (perhaps because they run in sequence and leave data for the next transaction), define a transaction group to include those three transactions specifying the affinity type and its duration.
2. If you want to implement workload separation, create a workload definition for each transaction group and user or terminal pattern, using the WLMDEF object. For example, if the intertransaction affinity described in item 1 (above) applies to any user from terminals NYORK*, the workload definition must include that information. In addition, the workload definition identifies the set of target regions from which CICSplex SM can select one. This could be every target region in the CICSplex, or some subset of target regions.

Note that you can define an intertransaction affinity to CICSplex SM without also implementing workload separation. In this case, no workload definition is required and CICSplex SM routes the first transaction in the group to a suitable target region from the set identified in the workload specification. Subsequent transactions in the transaction group are routed to the same target region while the affinity is active.

3. If the workload definitions are to be installed automatically when the first routing region in the workload starts, you should use the WLMGROUP object to:
 - a. Create a workload group.
 - b. Add the workload definitions to the workload group.
 - c. Associate the workload group with the workload specification.

You can also install a workload group or individual workload definitions manually for immediate effect.

Note: It's important that you define routing regions and target regions in groups, to ensure that changes to the composition of those groups are automatically reflected in CICSplex SM's workload management.

A recommended approach

When you decide to implement WLM functions in your enterprise, you should begin by defining the appropriate objects. You should:

- Use the WLMDEF view to create a workload definition.

a recommended approach

- Use the WLMGROUP view to create a workload group.
- Use the WLMSPEC view to create a workload specification.

Then, to create associations between these objects:

- Use the WLMDEF view to add the workload definition to the workload group.
- Use the WLMGROUP view to add the workload group to the workload specification.
- Use the WLMSPEC view to associate the workload specification with a CICS system or CICS system group.

You can use the MAP action command to display a visual map of the WLM definitions in your data repository. If you want to see a list of the objects and associations you have already defined, use the WLMINGRP, WLMINSPC, and WLMSCOPE views.

CICSplex SM includes a *Starter Set* of CICSplex SM object definitions that illustrates the concepts and components introduced in this chapter. The Starter Set is described in detail in Chapter 10, “The CICSplex SM Starter Set,” on page 101.

If you have problems defining any particular object or association, refer to the CICSplex SM Starter Set examples. The Starter Set includes many WLM examples that you can use as models for your own objects. You can copy the supplied samples directly into your data repository via the batched repository-update facility. For more information about using the Starter Set as a model see the *CICS Transaction Server for z/OS Installation Guide*. For information about the batched repository-update facility, see *CICSplex System Manager Administration*.

Where next?

Once you have identified your WLM requirements, you should define them to CICSplex SM as described in *CICSplex System Manager Managing Workloads*.

Chapter 7. Exception reporting using real-time analysis (RTA)

The chapter covers:

- “What is real-time analysis?”
- “System Availability Monitoring (SAM)” on page 75
- “MAS resource monitoring (MRM)” on page 76
- “Analysis point monitoring (APM)” on page 77
- “Managing RTA” on page 77
- “Planning for RTA” on page 79
- “Implementing RTA” on page 80
- “Where next?” on page 82

What is real-time analysis?

CICSplex SM's real-time analysis (RTA) function provides automatic, external, notification of conditions in which you have expressed an interest.

Real-time analysis is not concerned solely with commonly recognized error conditions: you can ask to be notified about any aspect of a resource's status.

For example, if you tell CICSplex SM that all files used by CICS systems AORPAY1 must be in ENABLED status, CICSplex SM sends out warning messages when the status of any of those files is other than ENABLED. However, CICSplex SM isn't restricted to telling you about a problem only after it has occurred. CICSplex SM's real strength lies in its ability to detect *potential* problems with CICS resources. Indeed, CICSplex SM is unrivaled in its ability to detect and warn of the slightest change in the status of a CICS resource. For example, CICSplex SM can tell you that resource contention is growing, or that the dynamic storage area (DSA) free space is falling, or that the number of users of a particular transaction is high: these can all be indicators of future problems that you could avoid by taking preventive action.

The notification can take the form of a console message, or of a generic alert to NetView, or both. The CICSplex SM application programming interface (API) makes it possible for automation products actively to gather status information from CICSplex SM, rather than waiting for CICSplex SM to send out MVS messages and SNA generic alerts.

Why automate?

The CICSplex SM operator can monitor RTA messages and take appropriate action. For example, the CICSplex SM operator can change the status of a file from DISABLED to ENABLED using the CICSplex SM FILED view. However, this approach is becoming less satisfactory as CICS availability and performance requirements grow. Most enterprises are using more and more CICS systems, often via cloning, to process a growing workload. This factor alone has made the CICSplex SM operator's task more difficult and error prone. At the same time, however, service-level agreements are setting performance and availability targets that are more demanding than ever before. Consequently, the ideal environment is one in which the CICSplex SM operator is presented with problems only in the most exceptional circumstances. Such an environment is now possible, because most actual and potential problems detected by CICSplex SM can be corrected or prevented by intelligent use of automation products, such as:

- NetView
- Automated Operations Control/MVS (AOC/MVS)

- Automated Operations Control/MVS CICS Automation (AOC/MVS CICS Automation)
- CICSplex SM itself. (Not only does CICSplex SM provide detailed and timely information about CICS resource status to other automation products, it is also capable of correcting many of the problems it detects.)

The RTA external notifications

An RTA external notification is generated when the status of a CICS resource is not as expected. The notification can take the form of either an *external message* or an *SNA generic alert*, or both.

External messages

External messages are MVS/ESA operator messages that are written to the MVS console by default. External messages can also be viewed from the CICSplex SM EVENT view while a condition is active. When a condition has been resolved, it disappears from the EVENT view. (However, external messages written to the EVENT view are also written to the CICSplex SM EYULOG to provide a record of the contents of the EVENT view.) In the EVENT view, external messages are sorted in order of urgency, with the message about the most urgent condition at the top of the list. The urgency of a condition is user-defined.

External messages are issued in pairs: one message is issued at the start of a condition, and another is issued when the condition ends. The messages, which all have the prefix “EYUPN”, are described in *CICSplex System Manager Messages and Codes*. External messages can be intercepted by automation products, such as NetView, AOC/MVS, and AOC/MVS CICS Automation.

SNA generic alerts

CICSplex SM can construct SNA generic alerts and transmit them, via the NetView program-to-program interface (PPI), to NetView for processing. NetView can trap SNA generic alerts in its automation table, which can either take action itself to resolve the problem, or start some automation code from a command list (CLIST) or command processor.

RTA functions

The RTA functions are:

- System Availability Monitoring (SAM): see “System Availability Monitoring (SAM)” on page 75
- RTA resource monitoring, which includes:
 - MAS resource monitoring (MRM): see “MAS resource monitoring (MRM)” on page 76
 - Analysis point monitoring (APM): see “Analysis point monitoring (APM)” on page 77

Advantages of RTA

The advantages of using RTA are:

- Actions can be taken within CICS, removing the requirement to use external automation products.
- Potential problems are detected and corrected before they become critical.
- Performance to the end-user is improved by the early detection of bottlenecks that could affect response time.
- The volume of information presented to operators is reduced by filtering out unnecessary messages.

System Availability Monitoring (SAM)

CICSplex SM's system availability monitoring (SAM) function monitors CICS systems during their planned hours of availability. If any of a set of predefined conditions occurs while a CICS system is being monitored, CICSplex SM sends out external notifications at the start of the condition and also when it is resolved. The conditions are:

SAMOPS

The CICS system (CICS TS,) is not available.

SAMSOS

A CICS/ESA dynamic storage area (DSA) is short on storage.

SAMSDM

A CICS/ESA system dump is in progress.

SAMTDM

A CICS/ESA transaction dump is in progress.

SAMMAX

CICS/ESA or CICS/MVS has reached maximum tasks.

SAMSTL

One or more tasks in a CICS/ESA or CICS TS system are stalled because of contention for resources such as DBCTL, DL/I, transient data, files, journals, and temporary storage .

For example, if CICS system AORPAY2 must be active from 0900 hours to 1200 hours and CICSplex SM finds it inactive at 0930 hours, CICSplex SM issues external messages by default. A typical response, from the CICSplex SM operator or from an automation product, would be to try to restart the CICS system, or perhaps to start another.

Customizing the SAM external notifications

When any of the SAM conditions occurs, CICSplex SM issues an external message by default. Each of the SAM conditions has a default severity and priority value that determines the position of the external message, relative to other external messages, in the EVENT view. For example, the SAMTDM condition has a severity of HW (High Warning) and a priority of 128, whereas the SAMSOS condition has a severity of HS (High Severe) and a priority of 255 (the highest priority), so SAMSOS precedes SAMTDM in the EVENT view.

You can customize the default SAM notifications for any CICS system by creating a CICSplex SM *action definition*. Using an action definition, you can change:

- The type of notification that is issued for a SAM condition. For example, you could specify that, for a particular CICS system, the SAMSOS condition will cause an SNA generic alert to be sent to NetView (so that NetView can take corrective action), and that no external message is to appear in the EVENT view.
- The severity value and the priority value of a SAM condition. For example, you could specify that the SAMTDM condition must have a severity of LW (Low Warning) in a particular CICS system, and that the SAMSOS condition must have a priority of 128.

You can also use the action definition to request that CICS TS systems affected by a condition are to be cancelled and restarted using the MVS automatic restart manager (ARM). This might be a suitable automation action for the SAMOPS condition, for example.

The default severity and priority values of the SAM conditions, and instructions for changing them using a CICSplex SM action definition, are included in *CICSplex System Manager Managing Resource Usage*.

Automation and SAM

An automation product, such as AOC/MVS or NetView, can intercept SAM notifications and take corrective action, such as starting up or shutting down a CICS system, disabling a transaction, or requesting a dump. For example, if a CICS system becomes unavailable, one of the automation products could attempt to restart the system, or perhaps start another.

However, most of the SAM notifications relate to actual problems that are already having an effect on users, so your scope for preventive action is small. Consequently, the best automation option for most SAM conditions is usually to request a dump, which can provide more information about a problem and its causes.

MAS resource monitoring (MRM)

Using CICSplex SM's MAS resource monitoring (MRM) function, you can monitor the status of *any* specific or generic CICS resource, and be informed when its status deviates from a specified norm. (SAM, by contrast, is the monitoring and evaluation of a “default” resource, the CICS system itself. Although you can tailor the type of notification you receive from SAM, the condition that triggers the notification is defined by CICSplex SM.) Using MRM, you can select both the resource status you are interested in and the type of external notification it generates. Here are some examples of the ways in which MRM can be used:

- You can use MRM to inform you of changes in the absolute status of a resource. For example, if the status of an LU6.2 connection is ever RELEASED at a time when you have specified it must be ACQUIRED, external notifications can be issued so that the CICSplex SM operator or an automation product can attempt to reacquire the connection. MRM can warn you of changes in the absolute status of many resources. For example, you can ask to be informed when a journal is CLOSED, when a transient data queue is DISABLED, when a FEPI node or a terminal is OUTSERVICE, when a transaction is DISABLED, and so on.
- In addition to monitoring absolute status values of CICS resources, CICSplex SM can provide information about subtle changes in a resource status, such as degradations in the response time of a transaction, or increases in the number of users of a program, or changes in the number of DB2® threads in a CICS system. These are all examples of trends in resource behavior that might be indicators of incipient problems.
- Using MRM, you can specify complex conditions. For example, you can ask for an external notification to be issued when the number of users of a particular transaction reaches a specified level *and* the dynamic storage area (DSA) free size is falling. Only when both conditions are true is the notification issued. Conditions can be of any complexity.
- MRM even supports the monitoring of non-CICS resources, such as DB2, and of other members of the CICS family, such as CICS/400, by invoking user-written programs called *status probes*.

As its name suggests, MAS resource monitoring operates at the CICS system level. That is, if a resource's status changes in two regions, two sets of external notifications are issued, one for each region.

Automation and MRM

MRM offers great scope for automation of CICS resource management, because it can alert you to both actual and potential problems. In alerting you to actual problems, MRM provides you with an opportunity to take *remedial* action; in alerting you to potential problems, MRM provides you with an opportunity to take *preventive* action. Furthermore, an automated response is possible to all MRM-detected problems.

When changes in the absolute status of a resource occur (for example, when a file that should be ENABLED becomes DISABLED, or when a connection that should be INSERVICE goes OUTSERVICE), remedial action is usually required. CICSplex SM itself can correct the status of such resources automatically, as can any of the automation products, such as NetView or AOC/MVS CICS Automation.

When more subtle changes in a resource's status occur, an automation product such as NetView can tailor its response to the severity of the problem. For example, CICSplex SM could monitor the DSA free size in a CICS system every five seconds. If CICSplex SM finds the DSA free size to be:

- Between 86KB and 100KB 4 times in succession, it issues an SNA generic alert to NetView with a severity of LW (low warning)
- Between 71KB and 85KB three times in succession, it issues an SNA generic alert to NetView with a severity of LS (low severe)
- 70KB or less twice in succession, it issues an SNA generic alert with a severity of VLS (very low severe).

In response to any of these alerts, the NetView automation table could issue an MVS modify command to the affected region to increase the DSA size dynamically, thereby preventing a short-on-storage condition (the SAMSOS condition, in fact) from occurring. The amount by which the DSA is increased in each case could vary according to the severity attached to the SNA generic alert by CICSplex SM.

Analysis point monitoring (APM)

CICSplex SM's analysis point monitoring (APM) function is the same as its MRM function, except that when resources are monitored in multiple CICS systems (a CICS system group or an entire CICSplex), the occurrence of the same problem in any number of those CICS systems can result in one external notification rather than several. APM is especially useful in environments that use cloned AORs, where regions are identical and one notification is sufficient to alert you to a general problem. APM does not support the use of status probes for monitoring non-CICS resources. In all other respects, however, APM offers the same opportunities as MRM for early detection of potential problems, and for an automated response in such situations.

Managing RTA

RTA is set up and managed using the objects described in Table 5.

Table 5. Real-time analysis objects

| Object name | Description | Function |
|-------------|--|---|
| APACTV | Analysis definitions associated with an analysis point specification | This object shown information about analysis definitions associated with an analysis point specification. |

Table 5. Real-time analysis objects (continued)

| Object name | Description | Function |
|-------------|--|---|
| ACTNDEF | Action definitions | An action definition designates the type of external notification that is to occur when the condition or conditions identified in an analysis definition are true. |
| APCMAS | Analysis point specification and CMAS association | This object shows the relationship between analysis point specifications and CMASs. During real-time analysis initialization, the association between analysis point specifications and CMASs is used to determine which specification should be installed within the CMAS in which real-time analysis is activated. |
| APSPEC | Analysis point specifications | This object shows information about analysis point specifications. An analysis point specification identifies one or more CMASs that are to be responsible for analyzing CICS systems within the CICSplex identified as the context. |
| EVALDEF | Evaluation definition | This object shows information about evaluation definitions. An evaluation definition identifies the resources in one or more CICS systems that are to be sampled and evaluated. When the result of the evaluation is true, an associated analysis definition is used to determine if a notifiable condition has occurred. |
| EVENT | Changes in the status of a CICSplex or one of its CICS systems | This object shows information about outstanding changes in the status of a CICSplex or one of its CICS systems. |
| EVENTDTL | Evaluation definitions associated with analysis definitions | This object shows information about evaluation definitions associated with an analysis definition that has caused an event to be generated. |
| RTAACTV | Analysis and status definitions | This object shows information about analysis and status definitions installed in CICS systems known to the CICSplex identified as the current context. |
| RTADEF | Analysis definitions | An analysis definition identifies the evaluations to be performed on a periodic basis and the actions to be taken should a notifiable condition occur. |
| RTAGROUP | Analysis groups | An analysis group is used to associate one or more related analysis definitions, status definitions, or both. |
| RTAINAPS | Analysis groups in analysis point specifications | This object displays the names of analysis groups associated with analysis point specifications. |
| RTAINGRP | Analysis and status definitions in analysis groups | This object displays the names of analysis groups and the analysis and status definitions associated with them. |
| RTAINSPC | Analysis groups in analysis specifications | This object shows the names of analysis specifications and the analysis groups associated with them. |
| RTASCOPE | Analysis specifications assigned a scope | This object shows information about the CICS systems or CICS system groups that are associated with an analysis specification. |
| RTASPEC | Analysis specifications | An analysis specification identifies the default control attributes that are used for system availability monitoring and provides an anchor for all analysis definitions and status definitions associated with a CICS system. |
| STATDEF | Status definitions | A status definition identifies a user-program that is to be called by CICSplex SM at specific intervals. |

The RTA object model is shown in Figure 20.

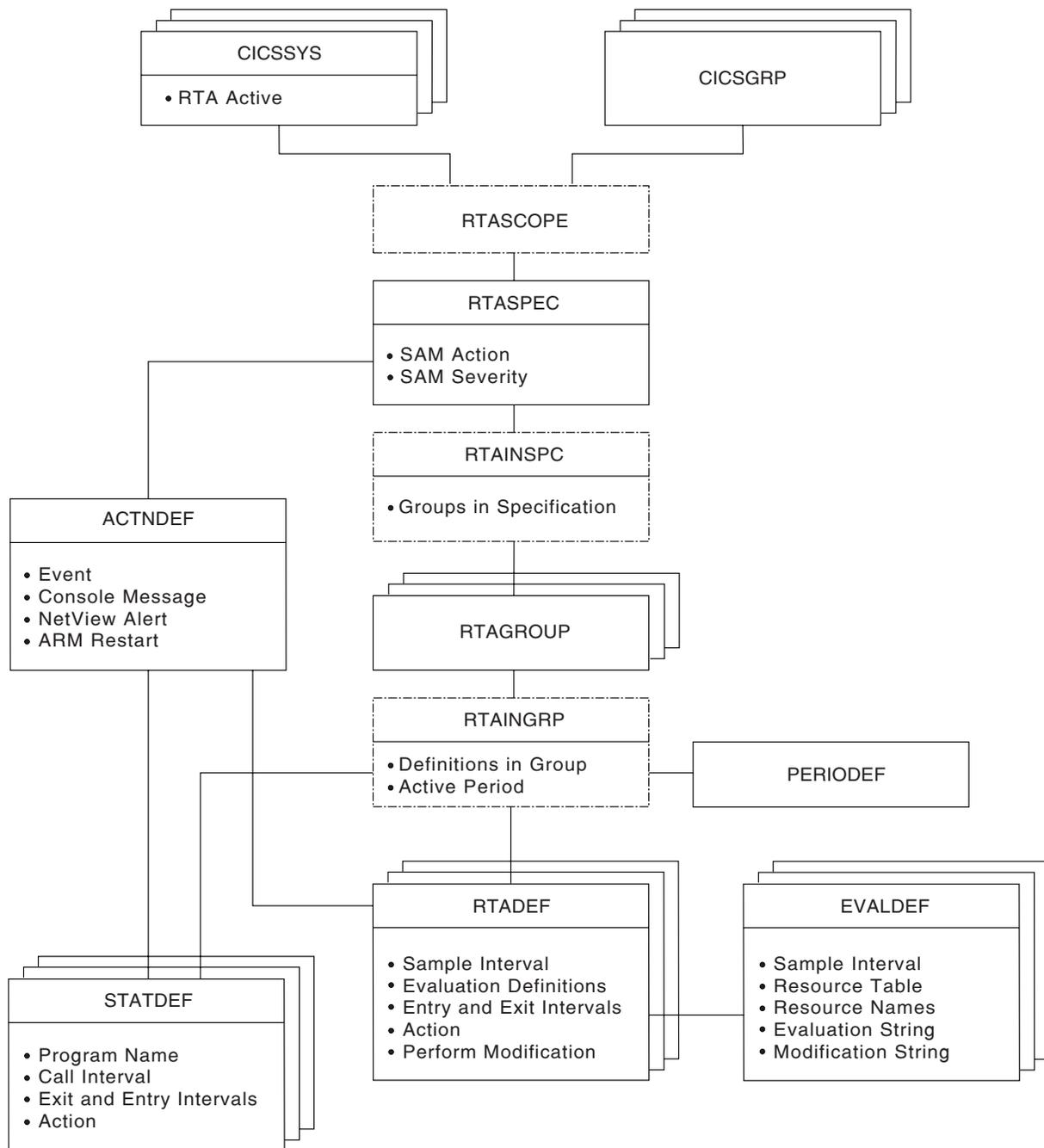


Figure 20. The real-time analysis object model

Planning for RTA

When you are planning to implement RTA, you might consider the following points:

- You will probably want to use SAM, because not only is it very easy to implement (see “Implementing system availability monitoring (SAM)” on page 80) but you will need to be aware when any of your CICS systems are not active when they should be. You need to:
 - Decide whether you want to customize any of the default conditions

- Identify the primary CMAS for each CICS system
- To use MRM and APM, you need to identify:
 - The resources you are interested in
 - The point at which their status becomes of interest to you
 - How you want to be notified of an interesting status
 - The interval at which you require resource evaluations
 - The interval at which you require data analysis
 - The primary CMAS

Note: For APM only, you may also specify one or more secondary CMASs.

Implementing RTA

Real-time analysis should be implemented gradually and in response to an established need for the resultant information, especially if you are not planning to automate your responses to exception conditions. You are recommended to begin by implementing SAM, and to implement MRM after having analyzed records, such as problem logs, that give you some indication of where MRM would be useful.

Implementing system availability monitoring (SAM)

You are recommended to implement SAM as follows:

1. Create time-period definitions (PERIODEFs) for the expected availability times of each CICS system to be managed by CICSplex SM. For example, if seven of your CICS systems should be active from 0800 hours through 1900 hours, create one time-period definition for that part of the day.
2. When you define each CICS system to CICSplex SM, supply the name of the time-period definition that defines the system's hours of availability.

These two actions are sufficient to activate SAM and to cause the default external notifications to be generated when a noteworthy condition occurs. As you can see, you can implement SAM as soon as CICSplex SM is installed in your enterprise, and with very little additional effort. You are recommended to continue using SAM with the default notifications, at least until you have had an opportunity to see the default notifications of all SAM conditions. If you decide to customize any of the notifications, start by creating an action definition for one of the SAM conditions and apply it to a single CICS system to test the effects. When you are happy with the change, apply it to other SAM conditions and other CICS systems, as appropriate.

The overheads associated with SAM are not high, so you can activate it throughout the enterprise CICSplexes without affecting the general performance of CICSplex SM or the CICS systems it manages.

Implementing MAS resource monitoring (MRM)

You should start by implementing MRM on a system-by-system basis. Select a single resource type, such as connections or files, in a single CICS system, and create these CICSplex SM objects:

- An *evaluation definition*, to identify the instances of the resource you are interested in, the frequency with which CICSplex SM is to evaluate the resource status, how CICSplex SM is to interpret its findings, and the severity to be assigned to a true condition

- An *analysis definition*, to identify the frequency with which CICSplex SM is to analyze the results and the point at which an external notification should be issued
- An *action definition*, to identify the type of external notification to be issued

The interval between resource evaluations (as specified in the evaluation definition) and the intervals at which data is to be analyzed (as specified in the analysis definition) should be intelligently matched (and ideally the same). In particular, do not analyze the data more frequently than it is evaluated. Also, resource evaluations should be performed as infrequently as possible while still providing useful data: the more frequently data is collected, the greater the cost of its collection.

Begin by installing the analysis definition manually in the active CICS system. When you are satisfied that MAS resource monitoring is working as you expected, consider whether automatic installation of the analysis definition would be useful. If it would, add the analysis definition to an analysis group, and associate the analysis group with an analysis specification. At this point, decide whether to monitor additional resources in the same CICS system. To extend MAS resource monitoring to other resources, an additional evaluation definition would be required, an additional analysis definition might be required, but the original action definition could probably be reused.

Implementing analysis point monitoring (APM)

When you are happy with the MAS resource monitoring results, progress to analysis point monitoring (APM), which is applied to a logical grouping of CICS systems. You should consider which CICSplexes are affected by your analysis point monitoring requirements, and identify the CMASs involved in management of those CICSplexes. As most of the CICSplex SM definitions required for MAS resource monitoring can be reused for analysis point monitoring, the cost of staging the implementation in this way is not high. For example, a single action definition is likely to be usable by multiple analysis definitions. However, for analysis point monitoring you must create at least one analysis point specification. As in the case of MAS resource monitoring, you should start analysis point monitoring with simple definitions, and move gradually to using more complex definitions.

A recommended approach

When you decide to implement RTA functions in your enterprise, you should begin by defining the appropriate objects. You should:

- Use the RTADEF view to create an analysis definition.
- Use the EVALDEF and ACTNDEF views to create the required subcomponents of an analysis definition.
- Use the RTAGROUP view to create an analysis group.
- Use the RTASPEC view to create an analysis specification.

Then, to create associations between these objects:

- Use the RTADEF view to add the analysis definition to the analysis group.
- Use the RTAGROUP view to add the analysis group to the analysis specification.
- Use the RTASPEC view to associate the analysis specification with a CICS system or CICS system group.

a recommended approach

You can use the MAP action command to display a visual map of the real-time analysis definitions in your data repository. If you want to see a list of the objects and associations you have already defined, use the RTAINGRP, RTAINSPC, and RTASCOPE views.

CICSplex SM includes a *Starter Set* of CICSplex SM object definitions that illustrates the concepts and components introduced in this chapter. The Starter Set is described in detail in Chapter 10, “The CICSplex SM Starter Set,” on page 101.

If you have problems defining any particular object or association, refer to the CICSplex SM Starter Set examples. The Starter Set includes many real-time analysis examples that you can use as models for your own objects. You can copy the supplied samples directly into your data repository via the batched repository-update facility. For more information about using the Starter Set as a model and about the batched repository-update facility, see the *CICS Transaction Server for z/OS Installation Guide* and *CICSplex System Manager Administration*, respectively.

Where next?

Once you have identified your RTA requirements, you should define them to CICSplex SM as described in *CICSplex System Manager Managing Resource Usage*.

Chapter 8. Collecting statistics using CICSplex SM monitoring

This chapter covers:

- “What is monitoring?”
- “Monitoring of resources”
- “The monitor interval” on page 84
- “The sample interval” on page 84
- “Managing monitoring functions” on page 84
- “Planning CICSplex SM monitoring” on page 86
- “Implementing CICSplex SM monitoring” on page 88
- “Where next?” on page 90

What is monitoring?

CICSplex SM *monitoring* supports the collection of performance-related data, at user-defined intervals, for named resource instances within a set of CICS systems. At user-defined intervals, monitoring gathers status information and statistics for selected resources, wherever they exist in the CICSplex, and makes them available at any point of control. CICSplex SM gathers information using standard CICS interfaces (EXEC CICS INQUIRE, EXEC CICS COLLECT STATISTICS) and can be used instead of the CICS Monitoring Facility (CMF). If CMF is running, CICSplex SM exploits the information provided. Monitoring need not be specified for resources being overseen by real time analysis (see “Implementing RTA” on page 80), as these are monitored separately.

Monitoring functions

The CICSplex SM monitoring functions are:

- Monitoring of resources: see “Monitoring of resources”
- Monitoring interval: see “The monitor interval” on page 84
- Sampling interval: see “The sample interval” on page 84

Advantages of monitoring

CICSplex SM monitoring offers the following advantages:

- Summary and detailed information on resources is readily available, enabling you to provide an improved help-desk service.
- System availability and performance are improved by allowing operators to be more effective in investigating problems and seeking out bottlenecks.

Monitoring of resources

CICSplex SM’s monitor functions can monitor the performance of these CICS resources at specified intervals:

- CICS systems
- Connections
- DB2 threads
- FEPI connections
- Files and data tables
- CICS journals
- Programs
- Temporary storage
- Terminals
- Transactions

monitoring of resources

- Transient data queues

The data collected by this monitoring activity is either the type of data that is collected by the CICS command EXEC CICS COLLECT STATISTICS, or is *derived* from such data, that is, CICSplex SM calculates rates, totals, averages, and percentages from the raw statistics data. For local MASs only, CICSplex SM can also obtain some detailed monitoring data via the CICS monitoring facility (CMF).

The monitor interval

When you monitor the activity of a particular resource, you are usually interested in its performance over a specific period of time. Therefore, when you define a CICSplex to CICSplex SM (using the CPLEXDEF view described in *CICSplex System Manager Administration*), you specify a *monitor interval* for the CICSplex. The monitor interval is a period of time, at the end of which counters holding derived monitoring data are reset to zero. For example, if you set the monitor interval to 60 minutes, counters are incremented for 60 minutes, then are reset to zero and the accumulation of monitoring data starts again. This sequence is repeated while monitoring is active. The interval is specified in minutes (15 through 1440), and defaults to 480 minutes (8 hours). Values derived by CICSplex SM during the monitor interval are prefixed with the letters “MI” in relevant views.

The sample interval

For each type of resource you monitor (such as files or connections), you specify a *sample interval*. The sample interval governs the frequency with which CICSplex SM collects performance-related data for selected instances of that resource type. For example, you can specify a sample interval of 300 seconds for files used in a particular group of CICS systems. Values derived by CICSplex SM during the latest sample interval are prefixed with the letters “CS” in relevant views. Thus, the CS values allow you to identify rapid rates of change, whereas the MI values are more useful in identifying longer-term trends.

Managing monitoring functions

The monitoring functions are set up and managed using the objects described in Table 6.

Table 6. Monitoring objects

| Object name | Description | Function |
|-------------|---|--|
| MONACTV | Monitor definitions installed in CICS systems | This object shows information about monitor definitions installed in CICS systems known to the CICSplex identified as the current context. |
| MONDEF | Monitor definitions | A monitor definition qualifies a resource type identified in a monitor specification by indicating the occurrences of the resource that are to be included in or excluded from monitoring. A monitor definition also identifies which resources are to be reported to the resource status facility when it is activated. |
| MONGROUP | Monitor groups | A monitor group is used to associate one or more related monitor definitions. |
| MONINGRP | Monitor definitions in monitor groups | This object displays the names of monitor groups, the monitor definitions associated with them, and the time period during which the monitor definitions are to be active. |

Table 6. Monitoring objects (continued)

| Object name | Description | Function |
|-------------|--|---|
| MONINSPC | Monitor groups in monitor specifications | This object displays the names of monitor specifications and the monitor groups associated with them. |
| MONSCOPE | Monitor specifications assigned a scope | This object displays information about the CICS systems or CICS system groups that are associated with a monitor specification. |
| MONSPEC | Monitor specifications | A monitor specification identifies one or more resource types that are to be monitored by CICSplex SM. |

The monitoring object model is shown in Figure 21 on page 86.

planning CICSplex SM monitoring

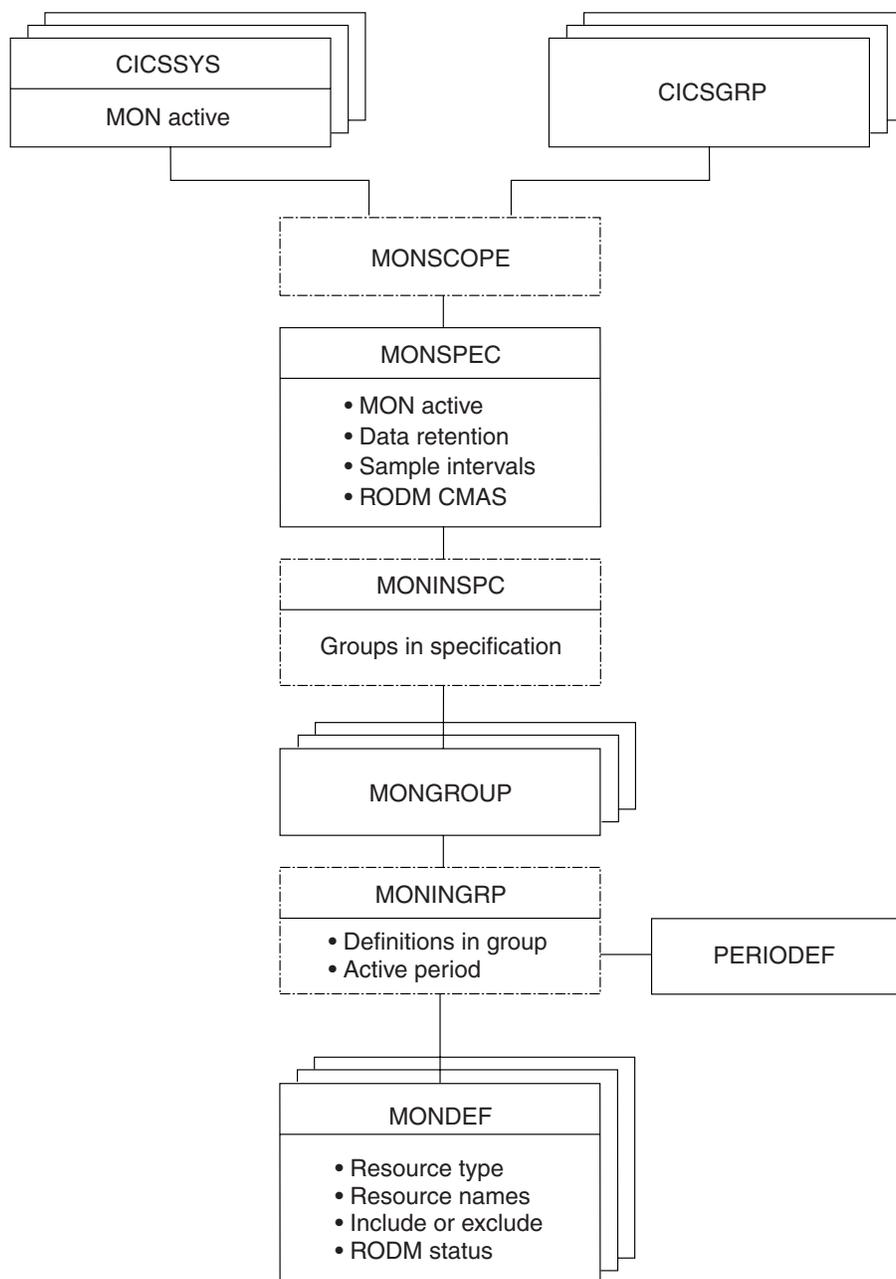


Figure 21. The resource monitoring object model

Planning CICSplex SM monitoring

This section provides some recommendations that will help you plan how to make the best use of CICSplex SM's monitoring functions in your enterprise.

Always have a use for the monitor data you request

Each resource class with a sample interval greater than zero has a CICS task in each MAS (that is, in each managed CICS system) for which monitoring is required. So, for example, if you activate monitoring of transactions in every CICS system in a CICSplex, each MAS is supporting a CICS task to monitor transactions at the specified interval. Also, CMAS processing is required to maintain the monitor-data cache. You can see that general and unrestricted monitoring would very quickly become an

unacceptable overhead. Therefore, do not activate monitoring for any CICS system nor for any CICS resource class without being aware of why you want the data.

Identify resources as specifically as possible

CICSplex SM monitoring is a specialized function that provides detailed usage and performance data. In general, it will be most useful to you for short-term trend analysis, in anticipating future problems, or dealing with actual problems. For example:

- Resources that are limited in quantity can be candidates for monitoring if their use levels are high, because the monitor data will help you to decide whether to increase quantities of the resource.
- Resources for which there is contention are also good candidates. Such data can help you to assess the impact of contention on throughput.

However, you should give some thought to understanding which resources are of greatest interest to you. For example, are you really interested in CICS/ESA supplied resources? Often it's tempting to supply an asterisk (*) as the resource name, which effectively means any instance of the specified resource. If you really aren't interested in every occurrence, use generic name patterns. For example, you can ask for monitoring of all transactions beginning "PAY" by specifying the resource name "PAY*". Similarly, resources whose status is being reported to the resource status facility should be identified as fully as possible.

Specify as long a sample interval as you can

The shorter the sample interval you specify for a resource class, the greater the cost of gathering the monitoring data. For example, if you set the sample interval to one second for every resource, no other CICSplex SM work is going to get done at all. Therefore, always specify the longest possible sample interval at which you can obtain useful data. To determine exactly what this interval should be, you must apply what you already know about any particular resource instance. In general, the sample interval should reflect the probable level of use of the resource: if you find that successive sampling of a resource is not showing significant changes, you should increase the sample interval. The sample interval should also have some relationship to the frequency with which the CICSplex SM or NetView operator will be looking at the monitor data.

Specify as long a monitor interval as you can

The monitor interval determines the frequency with which counters holding accumulated monitor data are reset to zero. You should set it to a value that is significant in your enterprise, so that the information you get from it is also significant. For example, the interval could be a shift duration, or some other time during which a particular workload is being processed.

You are recommended not to make the interval too short, not only because CICSplex SM's overheads grow as the monitor interval reduces, but also because too short an interval is unlikely to provide you with useful statistics.

Do not define more monitor specifications than you really need

Avoid having too many monitor specifications, not only because of the extra work involved in creating them and associating them with CICS systems, but also because a CICS system can belong to only one monitor specification at a time. The more monitor specifications you define, the smaller the opportunity for establishing a general-purpose monitoring policy.

Specify a suitable retention period

You don't have to specify a retention period. However, the retention period

planning CICSplex SM monitoring

defaults to zero minutes, and so as soon as monitoring stops for any CICS system, that system's monitor data is lost. If some unexpected event occurs, the data could be invaluable, so setting the retention period greater than zero is advisable. Note, however, that if a CICS system stops, monitor data is lost when you restart the system regardless of any retention period you may have specified. In this case, it's important to look at the monitor data before you restart the CICS system.

You can set the retention period to any value between 1 and 1440 minutes (24 hours). You can specify a retention period on the monitor specification, in which case that period applies to every CICS system associated with the monitor specification. Alternatively, you can specify a retention period in the CICSplex SM CICS system definition.

Decide when to activate monitor definitions

When you add a monitor definition to a monitor group, you can specify the part of the day during which it is to be active. You should make considered use of this feature. For example, you probably will not want monitoring active overnight, or during any period when general use is low and no one is around to view the data, so ensure that monitor definitions are not in effect at those times. Also, if you're monitoring resources for which there is contention, think about when this contention occurs. Is it constant, or does it occur only between 0900 and 1100 hours? If the latter, limit your monitoring to that period.

Implementing CICSplex SM monitoring

Possibly the best way to implement CICSplex SM monitoring is to start with no monitoring at all, and to implement it piecemeal for particular CICS systems and CICS resource instances. That is:

1. Create a monitor specification, and associate it with every CICS system in the CICSplex. However, ensure that every sample interval remains at its default value of zero, and that monitor status remains set to "OFF". This is because the one thing you must have in place before you can gather any monitoring data is a monitor specification. If you follow these instructions, monitoring will be enabled, but not *activated*.
2. When you decide to monitor a particular resource instance, as used by one of the CICS systems in the CICSplex, you must:
 - a. Update the CICS system definition to switch monitoring on, and to specify a sample interval for the resource. You make this change to an active CICS system (using the TOPOLOGY MAS view), and it applies for the current execution of the CICS system only.
 - b. Create a monitor definition to name the resource instance, and install it manually in the CICS system.
3. Repeat this piecemeal approach to monitoring until it becomes apparent that you have some regular requirements. At that point, consider installing the monitor definitions automatically (and for every execution of the CICS system) by adding the monitor definitions to a monitor group, and associating the monitor group with the monitor specification. You can make the sample interval and monitor status permanent by setting them either in the monitor specification—in which case, they apply to all CICS systems associated with the monitor specification and for which an appropriate monitor definition has been provided—or in the CICS system definition, in which case they affect no other CICS system.

You can refine your approach to monitoring gradually, by creating additional monitor specifications for the CICSplex, for example, and by using timed monitor definitions, which are installed automatically for a specified duration. Remember that:

- Although you can make changes to a monitor specification while it is in use, the effect of those changes is not felt until the CICS system is next started. Note, however, that you can use the MAS view to change monitoring values for an active CICS system.
- A CICS system can be associated with only one monitor specification at a time.
- You can associate a monitor specification with a CICS system that is already running, but the effect of that new association is not felt until the CICS system is next started.

Aim to ensure that any monitor specifications and monitor definitions you provide support the normal situation, and that CICS system overrides are used for the exceptions.

Activating the resource status facility

To activate the resource status facility you must:

- Update the CICSplex SM definitions of those CICSplexes whose resources are to be reported to the resource status facility.
- Update or create monitor definitions to set the resource status facility Status flag to YES and to identify resource instances.
- Update or create monitor specifications to set the sample interval for each resource type.

Instructions for all of these tasks are provided in *CICSplex System Manager Administration* and *CICSplex System Manager Managing Resource Usage*.

For more information about the resource status facility see the description of the RESSTATUS parameter in the *CICS Transaction Server for z/OS Installation Guide*

Using the CICS Monitoring Facility (CMF)

Some classes of monitoring data—in particular, the detailed task-related data that is displayed in the TASKD and MLOCTRAD views—cannot be collected unless CICS performance class data monitoring is active in the monitored CICS system. You can activate performance class data monitoring for an active system using the CICSplex SM operations functions or the CEMT SET MONITOR transaction. Alternatively, you can activate performance class data monitoring permanently via CICS SIT parameters.

CMF data is usually written to an SMF data set. However, if you prefer, you can collect CMF data for use by CICSplex SM without having it written to an SMF data set.

More information about the CICS SIT parameters and about suppressing the output of CMF data to an SMF data set is provided in the *CICS Transaction Server for z/OS Installation Guide*. For more information about CMF, see the *CICS/ESA Performance Guide*.

A recommended approach

When you decide to implement MON functions in your enterprise, you should begin by defining the appropriate objects. You should:

- Use the MONDEF view to create an analysis definition.

a recommended approach

- Use the MONGROUP view to create an analysis group.
- Use the MONSPEC view to create an analysis specification.

Then, to create associations between these objects:

- Use the MONDEF view to add the analysis definition to the analysis group.
- Use the MONGROUP view to add the analysis group to the analysis specification.
- Use the MONSPEC view to associate the analysis specification with a CICS system or CICS system group.

You can use the MAP action command to display a visual map of the MON definitions in your data repository. If you want to see a list of the objects and associations you have already defined, use the MONINGRP, MONINSPC, and MONSCOPE views.

CICSplex SM includes a *Starter Set* of CICSplex SM object definitions that illustrates the concepts and components introduced in this chapter. The Starter Set is described in detail in Chapter 10, “The CICSplex SM Starter Set,” on page 101.

If you have problems defining any particular object or association, refer to the CICSplex SM Starter Set examples. The Starter Set includes many resource monitoring examples that you can use as models for your own objects. You can copy the supplied samples directly into your data repository via the batched repository-update facility. For more information about using the Starter Set as a model and about the batched repository-update facility, see *CICS Transaction Server for z/OS Installation Guide* and *CICSplex SM Administration*, respectively.

Where next?

Once you have identified your monitoring requirements, you should define them to CICSplex SM as described in *CICSplex System Manager Managing Resource Usage*.

Chapter 9. Planning CICSplex SM installation and setup

This chapter provides information to help you to prepare for the installation of CICSplex SM in your existing environment. In particular, it describes:

- “Software, hardware, and storage requirements”
- “CICSplex SM naming conventions” on page 92
- “Security planning” on page 93
- “Defining time zones” on page 94
- “An overview of the installation process” on page 95
- “Preparing to set up a CAS” on page 95
- “Preparing to set up a CMAS” on page 96
- “Preparing to set up a MAS” on page 98
- “Reuse of CICSplex SM object definitions” on page 98
- “Where next?” on page 99

Instructions on installing CICSplex SM are given in *CICS Transaction Server for z/OS Installation Guide*.

Software, hardware, and storage requirements

This information is provided so that your enterprise can prepare an environment in which CICSplex SM can be successfully installed and operated. Note that this information is provided for guidance only, and that the *CICS Transaction Server for z/OS Program Directory* provides the most up-to-date information.

Prerequisite software release levels

The following products are either required in the environments in which CICSplex SM is installed, or may be installed in support of specific CICSplex SM functions but are not mandatory.

Release levels are minimum levels only: later levels (if any) can be used.

- CICS Transaction Server for OS/390 Version 1 Release 3 or later
- The IBM CICS Interdependency Analyzer for z/OS can be used if you need assistance in identifying inter-transaction affinities
- ISPF Version 3 Release 3 is required to support the CICSplex SM end-user interface
- MVS/DFP Version 3 Release 1 is required
- MVS/ESA:
 - MVS/ESA Version 5 Release 1 is required if the goal algorithm is to be used for WLM
 - MVS/ESA Version 5 Release 2 is required if the MVS/ESA automatic restart manager (ARM) function is required
- NetView:
 - CICSplex SM can send generic alerts to NetView Version 2 Release 3, or later
- RACF Version 2 Release 1 or equivalent external security manager is required
- TSO/E Version 2 Release 1 is required for support of ISPF

The CICSplex SM application programming interface (API) can be used in applications written in the following languages:

- Assembler

- PL/I
- COBOL
- C
- REXX (MVS Batch, TSO, and NetView)

Prerequisite hardware

The basic requirement is for a processor that supports the prerequisite operating system and has sufficient processor storage to meet the requirements of z/OS V1.4, CICS TS for z/OS, Version 3.1, the application programs, the access methods, and all other software being run. This includes the IBM eServer™ zSeries® 990.

Storage requirements

See *CICS Transaction Server for z/OS Program Directory* for the minimum storage requirements for CICSplex SM on an MVS image on which a CMAS and one or more MASs are defined.

CICSplex SM naming conventions

For its configuration and for its operation, CICSplex SM requires a variety of entities to be defined. You must devise a convention for naming those entities that is both meaningful and extendable. The rules governing the naming of CICSplex SM elements are as follows:

- Each element name can be up to 8 characters.
- The element name must begin with an alphabetic (or national) character, and must have no embedded blanks. Subsequent characters can be alphabetic or numeric.
- Names of CICSplexes and of CMASs must be unique within the enterprise.
- Names of CICS systems and CICS system groups must be unique within the CICSplex.
- All other names must be unique for their type within the CICSplex.

The entity types, for each instance of which a name is required, are:

- CICSplex SM components and CICSplex entities
 - CAS, CMAS, CICSplex, MAS (CICS system), CICS system group, and time-period definition.
- BAS entities
 - Resource assignments, resource descriptions, resource definitions, and resource groups.

Application resource definitions are:

- DB2 transactions, files, map sets, partition sets, programs, transient data queues, transactions, CICS BTS process types, document templates, FEPI nodes, FEPI pools, FEPI property sets, FEPI targets, file and key file segment definitions, IIOp request models, map sets, sysplex enqueue models, TCP/IP services, temporary storage models

Region resource definitions are:

- DB2 entries, enterprise beans, journals, journal models (CICS Transaction Server for OS/390, Version 1 Release 3 or later), local shared resource (LSR) pools, profiles, temporary storage queue models, transaction classes, terminals, typeterms

System–system connectivity resource definitions are:

- Connections, DB2 connections, partners, sessions

- Monitor entities
 - Monitor definition, monitor group, and monitor specification
- Analysis entities
 - Analysis definition, analysis group, analysis point specification, analysis specification, action definition, evaluation definition, and status definition
- Workload management entities
 - Transaction group, workload definition, workload group, and workload specification.

Ensure that any convention you select can accommodate all of these entity types. A naming convention, for use generally in CICS environments, is described in the CICS/ESA publication *Dynamic Transaction Routing in a CICSplex*. For an example, see “The Starter Set naming convention” on page 125.

Generic names

CICSplex SM supports the use of generic names in many of its commands. For example, you can specify that all transactions whose names begin with the letters “DNW” are to be monitored once every 300 seconds. You do not have to name each transaction individually.

The rules governing use of generic names are as follows:

- The asterisk character (*) can be used in place of one or more characters. It can appear once only in any name, and must appear at the end of any partial string. For example, the string “DNW*” means all names beginning with the letters DNW.
- The plus sign (+) can replace any single character and can occur in any position in the name. For example, “DNW+++L” means any name of 7 characters starting with the letters DNW and ending with the letter L.

Security planning

CICSplex SM uses a SAF-compliant external security manager, such as RACF:

- To prevent unauthorized access to CICSplex SM functions and CICS resources
- To control the simulation of CICS command checking and CICS resource checking

In both cases, security checking is handled by the CMASs managing the CICS systems that are the target of any request to access a resource. For example, if a CICSplex is managed by two CMASs, and a request is made to access a resource in all CICS systems belonging to that CICSplex, the security check is performed in both CMASs.

To activate security checking, you must modify the JCL used to start the CMAS or its managed CICS systems. If security checking is switched off for the CICS system, no checking occurs, regardless of the CMAS setting. However, if security checking is switched off for the CMAS but switched on for the CICS system, the CICS system is not able to connect to the CMAS.

You should begin by deciding how much security checking you need. In particular, identify those users who need access to CICSplex SM, and ensure that an individual user has the same user ID across all systems on which a CMAS is installed. The user ID against which the security check is performed is the signed-on TSO user ID. Consider also the type of security checking you want to implement.

Protecting access to CICSplex SM functions and CICS resources

To prevent unauthorized access, you create security profiles for combinations of CICSplex SM functions, and to CICS resources that are to be protected. A table of valid combinations is provided in *CICS RACF Security Guide*.

In most cases, the security provided by these CICSplex SM security profiles is adequate.

An external security manager is also used to protect CICSplex SM's own libraries, procedures and Web User Interface resources. Full details of how to protect CICSplex SM's libraries and procedures are provided in *CICS RACF Security Guide*. In order to protect Web User Interface views, menus, help information and the View Editor you need to create an appropriate profile in the FACILITY class. See the *CICSplex System Manager Web User Interface Guide* for more information.

Special considerations for BAS

You should be aware of the need to take special care in the protection of the BAS views, so that unauthorized users cannot create and administer resources. The equivalent in RDO terms is leaving your CSD unprotected.

You should also take care if you are using the EXEC CICS CREATE command to build new resources. Any definition created with the CICSplex as the context is automatically distributed to all CMASs in the CICSplex. Therefore, giving a user authority to create BAS objects is equivalent to giving authority to install resources on any CICS system in the CICSplex. When the CICS system starts, there is no check on who installed the resource in the system.

CICS command and resource checking

CICS command and resource checking is simulated by CICSplex SM in the CMASs to which a request is directed. This allows you to protect CICS systems that do not support your external security manager. It also allows for a level of consolidation of your security checking. Determine where CICS resource and command checking is in effect, and decide whether it needs to be retained along with CICSplex SM's other security checking.

Defining time zones

Much of CICSplex SM's activity is time dependent. For example, you can specify that a monitor definition or an analysis definition is to be active during a particular time period. CICSplex SM does not require every MAS in a single CICSplex to be running in the same time zone, and so must be able to accommodate any time-zone differences between entities. Therefore:

- Whenever you create a time-period definition (using the CICSplex SM PERIODEF view), you must specify a time zone in the definition. For example, you could create a time-period definition called "MORNING" for the hours 0800 through 1159 *Eastern standard time*.
- You must specify a time zone for each CMAS in its data-repository initialization job, EYU9XDUT. A permanent change to the CMAS time-zone value can be made, even while the CMAS is running, via the CICSplex SM user interface.
- A time zone must be established for each managed CICS system. When you define a CICS system to CICSplex SM, you can specify the time zone in which the system is running. Alternatively, if you don't specify a time zone in the CICS system definition, the CICS system is assumed to be running in the time zone

that is the default for the CMAS to which it connects. You are recommended to allow the time zone of a managed CICS system to default to that of its CMAS. The time zone of a managed CICS system can be altered subsequently while the CICS system is running. Any change made in this way lasts for the lifetime of the CICS system, or until it is next changed, whichever is sooner.

- A time zone must be specified for every CICSplex when it is first defined. This time zone is used by the CICSplex SM monitor function to determine the actual time at which the monitor interval for the CICSplex expires. The CICSplex time zone can be altered via the CICSplex SM end-user interface.

Time zones are specified using single-character codes in the range B through Z. For example, code “S” represents Mountain Standard Time, code “T” represents Central Standard Time, and code “C” represents Eastern Europe time. A complete list of the codes can be found in *CICSplex SM Administration*. CICSplex SM allows offsets (known as “time zone adjustments”) in the range 0 through 59 minutes to be specified to accommodate regions that are not running in the standard time zones. Also, daylight saving time can be specified.

Because multiple CICSplex SM entities require a time zone to be specified, there is obvious potential for “conflicting” time-zones to be specified. For example, it is quite possible that a CMAS and a MAS in the same CICSplex could be in different time zones. CICSplex SM always honors the time zone of the MAS. For example, if:

- The time-period definition time zone is S

and

- The CMAS time zone is B

and

- The MAS time zone is C

time zone C is used by the MAS, and the CMAS makes any necessary adjustments between time zones B, C, and S to ensure that the time zone is honored.

An overview of the installation process

For information about installation see, in the first instance, *CICS Transaction Server for z/OS Program Directory*. For information about post-installation and IVPs see *CICS Transaction Server for z/OS Installation Guide*.

Preparing to set up a CAS

To set up a CAS, you need to:

Decide how to start the CAS

The installation process for CICSplex SM creates some skeleton start-up JCL that can be used to create and initialize the CAS. Each CAS requires a unique, four-character subsystem ID that must be included in the CAS start-up JCL.

A CAS can be started:

- At MVS IPL time
- From the system console
- As a batch job

Although you have this choice, you are strongly recommended to start each CAS in your enterprise at MVS IPL time, because it avoids the sequencing problems that occur when components are started in an unexpected order.

Also, you should consider CICSplex SM to be an essential part of your environment, much as VTAM and JES are considered today.

Provide CAS VTAM definitions

If your system “map” shows more than one CAS, you need to define some links between those CASs. For this, each CAS must have a VTAM application definition. Also, each CAS can be defined as a cross-domain resource, which allows it to be referenced across a link. Details of these requirements can be found in *CICS Transaction Server for z/OS Installation Guide*

Decide how the CICSplex SM EUI is to be accessed

There are two ways of starting the CICSplex SM EUI:

1. You can start CICSplex SM as an application from an ISPF menu, which requires you to make some permanent changes to your existing ISPF environment. In particular, you must add some CICSplex SM libraries to the signon procedure, and update an appropriate ISPF menu to include CICSplex SM as an option. This is the recommended way of starting CICSplex SM.
2. You can start CICSplex SM from the TSO READY prompt by running the sample REXX EXEC that was created by the installation process. This requires no permanent changes to the current environment, but is a less efficient way of starting CICSplex SM. One reason for this relative inefficiency is that the current dataset allocation has to be saved, the required CICSplex SM libraries must be allocated, and the original data set allocation must be restored when the user has finished using CICSplex SM. This approach is not recommended for regular use of CICSplex SM, though it is a useful installation aid.

Preparing to set up a CMAS

The CMAS, which is a CICS Transaction Server for z/OS Version 3 Release 1 system, provides most of the functions of CICSplex SM. To set up a CMAS, you need to:

Add CICS resource definitions for the CMAS

Because the CMAS is a CICS system, it requires resource definitions to be created. These definitions are provided with CICSplex SM and can be applied using the resource definition utility, DFHCSDUP. The resources include:

- CICSplex SM resource groups
- A file definition for the data repository
- CMAS program definitions
- CMAS transaction definitions
- CMAS group list

No tailoring of these resources is required or expected. However, you are required to make some changes to the tables DFHPLT and DFHSRT. These changes are detailed in *CICSplex SM Managing Workloads*.

Define common data spaces

In each MVS/ESA environment in which a CMAS is to be installed, a minimum of five common data spaces (and a maximum of 45) is required. You are recommended to begin by defining 10 common data spaces, and to add more if necessary.

Provide CMAS VTAM definitions

If you are defining multiple CMASs, CMAS-to-CMAS links require each CMAS to have a VTAM application definition.

Full details of necessary changes to the MVS/ESA and VTAM environments are provided in *CICS Transaction Server for z/OS Installation Guide*.

Create the CMAS data repository

Each CMAS requires a data repository, which can be created using the sample JCL created during the installation process. The data repository must be initialized before it can be used.

Decide how to start the CMAS

Like the CAS, the CMAS can be started at MVS IPL time, as a started task, or as a batch job. You are recommended to start the CMAS at MVS IPL time, so that it is part of the MVS/ESA start-up procedure and treated as a permanent part of the environment. This is much more efficient than starting and stopping the CMAS throughout the day. It also ensures that the CMAS is active before any MAS attempts to connect to it. A sample start-up procedure for the CMAS is generated during the installation process.

Using the batched repository-update facility

CICSplex SM supplies a batched repository-update facility that you can use to make bulk changes to a CMAS's data repository. You can invoke this facility from the Web User Interface, the EUI, the CICSplex SM API or from a batch utility. When using the batched repository-update facility, you supply an input file of commands that are applied to the data repository. Valid commands are:

CONTEXT

which identifies whether subsequent commands relate to the CMAS whose repository is being updated, or to a CICSplex that it maintains.

CREATE

which adds a new record to the data repository.

UPDATE

which alters an existing record in the data repository.

REMOVE

which deletes an existing record from the data repository.

LIST

which creates a list of all records in the data repository.

DUMP

which creates a list of all records in the data repository and prefaces each entry in the list with a Create command. The output from the Dump command can therefore be used as input to a different CMAS data repository.

MAPLEFT

which displays a map of definitions in the data repository to the left of the starting point.

MAPRIGHT

which displays a map of definitions in the data repository to the right of the starting point.

For example, a typical command file for input to the batched repository-update facility would contain a Context command, followed by multiple Create commands. An example command file is supplied with the CICSplex SM Starter Set. Detailed information about the batched repository-update facility is supplied in *CICSplex SM Administration*.

Preparing to set up a MAS

This section introduces the actions you need to undertake when you set up a MAS. Full details are given in *CICS Transaction Server for z/OS Installation Guide*. You need to:

Define CICSplex SM's resources to the CMAS

The CICSplex SM code, for MASs, is a CICS application, so you must define CICSplex SM's resources to the managed CICS system. Resource definitions, and instructions for their application, are supplied for all CICS environments (CICS Transaction Server).

Define the CICS system to CICSplex SM

Use the **CICS system Definition** WUI view or the CICS SYS EUI view.

Modify the MAS's startup JCL

The startup JCL must refer to the appropriate CICSplex SM data sets and include the required CICS system initialization parameters are included.

Ensure that the CMAS is started before the MAS

If the MAS is active before the CMAS, the MAS-attach process waits until the CMAS is started. This means that no CICSplex SM function is available to the CICS system until the CMAS is available. (However, CICSplex SM's WLM functions are available in these circumstances if the workload has been activated.) Delays in the MAS-attach process do not affect the functions of the CICS system itself.

Reuse of CICSplex SM object definitions

CICS does not provide identical support for resources and functions across all of its platforms or releases. For example, a resource, such as FEPI, that is supported in the CICS TS environment might not be supported on other platforms. Similarly, the amount or type of data available for a resource can vary among CICS platforms and releases. These variations in support pose a special challenge to CICSplex SM, which offers a SSI of multiple and varied CICS systems.

Here are some examples that illustrate how CICSplex SM handles variations in support for resources and functions throughout a CICSplex that includes CICS TS systems.

- **Example 1: monitoring of transient data queues**

You create a monitor definition for transient data queues, add it to a monitor group, and associate the monitor group with a monitor specification. The scope of the monitor specification is every CICS system in the CICSplex. As each CICS system starts, CICSplex SM determines whether it can install the monitor definition in that system. If not, CICSplex SM issues a message to inform you that the monitor definition could not be installed. Both the monitor function and transient data queue resources are supported under CICS TS, so CICSplex SM installs the monitor definition in those environments and monitoring begins.

- **Example 2: RTA of monitored programs**

You create an evaluation definition and name MPROGRAM (monitored programs) as the resource to be evaluated by CICSplex SM's RTA function. The evaluation definition is named in an analysis definition, which is linked, via an analysis group, to an analysis specification. The scope of the analysis specification is every CICS system in the CICSplex. The analysis definition is installed in the CICS TS systems. If an analysis definition names multiple evaluation definitions, CICSplex SM installs those that are supported in the target environment.

In summary, if you request a function that isn't available in a particular CICS environment, CICSplex SM issues a warning message. You do not have to respond to the message and, more importantly, you do not have to take variations in support into account when creating CICSplex SM object definitions: you should always consider CICSplex SM object definitions to be reusable.

Where next?

At this point, you are ready to install CICSplex SM and run the JCL to install your CASs, CMASs, and MASs. You now need to define the objects that will allow these entities to communicate with CICSplex SM and each other. *CICSplex SM Administration* gives you guidance on defining the CICSplex SM configuration and topology.

Chapter 10. The CICSplex SM Starter Set

The CICSplex SM Starter Set is primarily a set of samples—sample JCL, sample CICSplex SM topology definitions, and sample specifications for business application services, workload management, real-time analysis, and monitoring—based on an authentic configuration of eight CICS regions across two MVS/ESA images. This chapter covers:

- “Why is the Starter Set provided?”
- “The Starter Set configuration”
- “CICSplex SM definitions supplied with the Starter Set” on page 104
- “The Starter Set naming convention” on page 125

Why is the Starter Set provided?

The CICSplex SM Starter Set is provided to give you:

- Samples (of both CICSplex SM definitions and JCL) that you can copy and adapt for your own environment. For example, there are more than 100 monitoring and RTA samples that you can use as a basis for your own definitions.
- An understanding of what’s involved in setting up a CICSplex SM configuration.
- An opportunity to use the batched repository-update facility with sample data.
- A system that will allow you to familiarize yourself with the CICSplex SM user interface before you start to use it in earnest.
- A system on which you can learn how CICSplex SM is used as a system-management tool.

It is not intended that you should use the Starter Set as an initial configuration to build on by replacing its data with your own. There is nothing to stop you doing this, of course, but, as the Starter Set configuration is unlikely to resemble your own to any significant extent, the benefits would be few. Also, if you update the Starter Set definitions with your own data, you will effectively lose the Starter Set. New users of CICSplex SM would not have the opportunity to learn from the Starter Set, and examples from the Starter Set, which are used throughout the CICSplex SM documentation, would be less valid. Therefore, you are recommended to retain the Starter Set as a separate, sample configuration to which you can refer at any time, but to copy anything (such as JCL or CICSplex SM definitions) that you want to adapt for local use.

The Starter Set configuration

The Starter Set contains examples of configuration relationships and of BAS, WLM, RTA, and monitoring specifications and definitions. You can use these examples to model your own definitions.

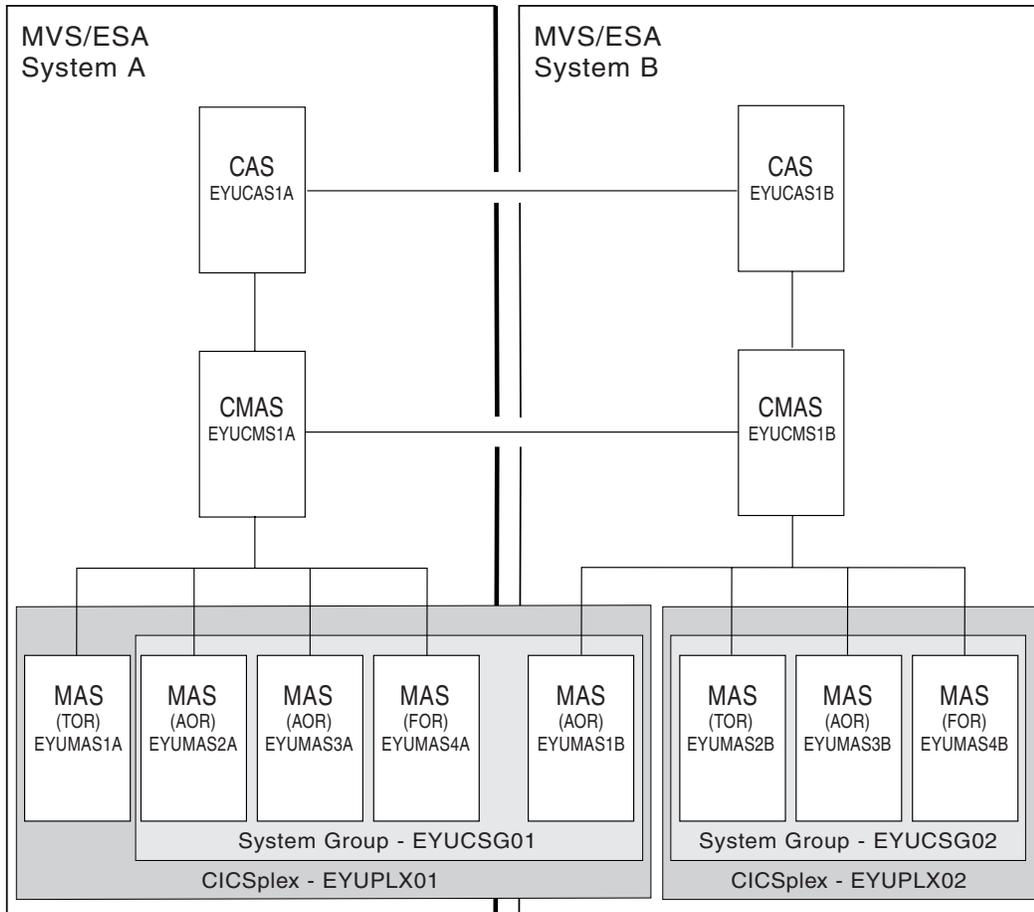


Figure 22. The Starter Set configuration. The Starter Set provides a small but comprehensive set of CICSPlex SM components on which you can model your own CICSPlex SM environment.

Figure 22 shows the main components of the CICSPlex SM Starter Set configuration, which consists of:

- Two CICSPlexes, EYUPLX01 and EYUPLX02, which straddle two MVS images. Neither CICSPlex is confined to a single MVS image. The Starter Set has two CICSPlexes rather than one because a configuration of multiple CICSPlexes allows all of CICSPlex SM's features to be demonstrated.
- Two CASs, one on system A and one on system B. Either CAS can function as a single point of control for all of the Starter Set MASs and their resources.
- Two CMASs, one on system A and one on system B. EYUCMS1A is the maintenance point CMAS for EYUPLX01, and EYUCMS1B is the maintenance point CMAS for EYUPLX02. The CMASs are connected to each other so that the CICSPlex SM data maintained by one CMAS is visible to the other. This ensures a single-system image for the CICSPlex SM user. Each CMAS has a data repository. (Data repositories are not shown in Figure 22.)
- Two CICS system group definitions, EYUCSG01 and EYUCSG02, that include a subset of the MASs in each CICSPlex. The Starter Set includes seven CICS system group definitions (EYUCSG01 through EYUCSG07), which are described in Table 10 on page 106 and in Table 13 on page 106.

The Starter Set installation requirements

The Starter Set is supplied in two samples libraries that are installed automatically when CICSPlex SM itself is installed.

The two libraries, whose default names are CICSTS31.CPSM.SEYUJCL and CICSTS31.CPSM.SEYUDEF, are described in the remainder of this section.

Starter Set library CICSTS31.CPSM.SEYUJCL

The samples library CICSTS31.CPSM.SEYUJCL contains sample JCL for creating and running the Starter Set CASs, CMASs, and MASs. It also includes JCL for creating the CMAS data repositories.

This JCL has been provided chiefly so that you don't have to create it yourself before you can use the Starter Set. However, you might also find it useful to refer to when you are creating JCL for your own CICSplex SM environment (for which you are recommended to use the skeleton JCL supplied with CICSplex SM).

CICSTS31.CPSM.SEYUJCL also includes sample JCL for deleting the Starter Set components from an MVS image.

Starter Set library CICSTS31.CPSM.SEYUDEF

The samples library CICSTS31.CPSM.SEYUDEF contains an assortment of sample definitions required to establish not only the Starter Set itself, but also the environment in which it can be used. (The data required to set up the environment is mostly of a routine nature, and is supplied primarily so that you don't have to define it yourself. However, as with the Starter Set JCL, you might find these samples to be a useful checklist or aid when you are making these changes in a real CICSplex SM environment.)

Definitions in CICSTS31.CPSM.SEYUDEF include:

- CICSplex SM definitions (for example, monitor specifications, time-period definitions, analysis definitions, CICS system group definitions, and workload definitions) for each of the two CICSplexes. These definitions are described beginning on page 104.
- VTAM definitions.
- Examples of changes to be made to the ISPF logon procedure and a sample initial menu.
- Updates for CICS tables (including the PLT and SRT).
- Additions to the CSDs for CMAS and MAS operation.
- Group-list definitions to identify the resources used by each MAS.
- Start-up parameters (EYUPARMs) for CMASs and MASs.

Configuring part of the Starter Set

The Starter Set configuration of seven CICS systems distributed across two MVS images was designed to demonstrate all of CICSplex SM's facilities. However, not having this exact environment doesn't mean that you can't use at least a part of the Starter Set. As an alternative to configuring the whole Starter Set in three MVS images, you can configure:

- The system A components only, on a single MVS image
- The system B components only, on a single MVS image
- The components of CICSplex EYUPLX01 (the system A and system B components of the Starter Set) on two MVS images
- The components of CICSplex EYUPLX02 (system B components of the Starter Set) on one MVS image

How to configure the Starter Set is described in the *CICS Transaction Server for z/OS Installation Guide*.

The Starter Set and the IVPs

Configuration of the Starter Set is not essential. However, part of the Starter Set constitutes the CICSplex SM MVS/ESA installation verification procedures (IVPs): if you run an IVP on an MVS image (as you are recommended to do), you will, during the IVP, perform most of the configuration tasks required to make the Starter Set usable. The IVPs ensure that:

- CICSplex SM has been installed successfully from the distribution tape
- All components of CICSplex SM are accessible
- Any MVS system tuning for CICSplex SM has been successful
- You are able to use the product

There are two IVPs, which are referred to as IVP1 and IVP2. IVP1 is run on the first (or only) MVS image on which CICSplex SM is installed and uses the system A components of the Starter Set. IVP2 is run on the second and subsequent MVS images on which CICSplex SM is installed. It uses both the system A and system B components of the Starter Set, and tests communication between CASs and CMASs. The IVPs are described in detail in the *CICS Transaction Server for z/OS Installation Guide*.

CICSplex SM definitions supplied with the Starter Set

The Starter Set includes many sample definitions for each of the CICSplex SM functions. That is, the Starter Set includes:

- CAS definitions, for CASs EYUCAS1A and EYUCAS1B
- Configuration definitions, for CMASs EYUCMS1A and EYUCMS1B and for CICSplexes EYUPLX01 and EYUPLX02
- Topology definitions, for the nine CICS systems in the Starter Set, for CICS system groups, and for time-period definitions
- BAS definitions, including resource definitions, resource groups, resource assignments, resource descriptions, and communication links for sample applications
- Workload definitions, including workload specifications, workload definitions, workload groups, and transaction groups
- Monitor definitions, including monitor specifications, monitor definitions, and monitor groups
- Analysis definitions, including analysis point specifications, analysis specifications, analysis groups, analysis definitions, evaluation definitions, action definitions, and status definitions

You add some of these definitions to the CMAS data repository interactively when you follow the instructions for configuring the Starter Set in the *CICS Transaction Server for z/OS Installation Guide*. However, you add most of them to the CMAS data repository using the CICSplex SM batched repository-update facility. A brief description of all of the Starter Set definitions is supplied in Table 7 on page 105 through Table 54 on page 124. For a more detailed understanding of these definitions, you are recommended to study the definitions as supplied in CICSTS31.CPSM.SEYUDEF.

Sample CAS definitions

Table 7 on page 105 lists the Starter Set CAS definitions that you create interactively using the CICSplex SM CAS functions. There are also links between

the CASs—from EYUCAS1A to EYUCAS1B, and from EYUCAS1B to EYUCAS1A—but such links are not separate entities in the CICSplex SM environment.

Table 7. Starter Set CAS definitions

| Type | Name | Description |
|------|----------|--------------|
| CAS | EYUCAS1A | System A CAS |
| CAS | EYUCAS1B | System B CAS |

Sample configuration definitions

Table 8 lists the Starter Set configuration definitions you create using the CICSplex SM Configuration functions.

Table 8. Starter Set configuration definitions

| Type | Name | Description |
|----------|----------|--|
| CMAS | EYUCMS1A | System A CMAS |
| CMAS | EYUCMS1B | System B CMAS |
| CICSplex | EYUPLX01 | CICSplex comprising all CICS systems (MASs) on system A and and EYUMAS1B on system B |
| CICSplex | EYUPLX02 | CICSplex comprising EYUMAS2B, EYUMAS3B, and EYUMAS4B on system B. |

The configuration definitions also include these links:

- From EYUCMS1A to EYUCMS1B
- From EYUCMS1B to EYUCMS1A

As in the case of the CAS-to-CAS links and CMAS-to-CMAS, links are not separate CICSplex SM entities.

Sample topology definitions

The Starter Set topology definitions include CICS system definitions, CICS system group definitions, and time-period definitions.

Topology definitions for EYUPLX01

Table 9 through Table 11 on page 106 list the Starter Set topology definitions created with EYUPLX01 as the context. Figure 23 on page 107 shows a Starter Set topology view.

Table 9. CICS system definitions for EYUPLX01

| Name | Belongs to | Description |
|----------|----------------------|--|
| EYUMAS1A | N/A | A test CICS Transaction Server for OS/390 Release 3 TOR on system A; its hours of availability are as defined by time-period definition EYUPDF01 |
| EYUMAS2A | EYUCSG01 EYUCSG04 | A test CICS Transaction Server for OS/390 Release 3 AOR on system A; its hours of availability are as defined by time-period definition EYUPDF01 |
| EYUMAS3A | EYUCSG01 EYUCSG04 | A test CICS Transaction Server for OS/390 Release 3 AOR on system A; its hours of availability are as defined by time-period definition EYUPDF01 |

Table 9. CICS system definitions for EYUPLX01 (continued)

| Name | Belongs to | Description |
|----------|------------|--|
| EYUMAS4A | EYUCSG01 | A test CICS Transaction Server for OS/390 Release 3 FOR on system A; its hours of availability are as defined by time-period definition EYUPDF01 |
| EYUMAS1B | EYUCSG05 | A test CICS Transaction Server for OS/390 Release 3 AOR on system B; its hours of availability are as defined by time-period definition EYUPDF01 |

Table 10. CICS system group definitions for EYUPLX01

| Name | Belongs to | Description |
|----------|------------|---------------------------|
| EYUCSG01 | N/A | AORs and FORs in EYUPLX01 |
| EYUCSG03 | N/A | All AORs in EYUPLX01 |
| EYUCSG04 | EYUCSG03 | System A AORs in EYUPLX01 |
| EYUCSG05 | EYUCSG03 | System B AORs in EYUPLX01 |

Table 11. Time-period definitions for EYUPLX01

| Name | Belongs to | Description |
|----------|------------|---------------------------|
| EYUPDF01 | N/A | 08:00 to 17:00 local time |
| EYUPDF02 | N/A | 17:01 to 21:00 local time |
| EYUPDF03 | N/A | 11:45 to 13:15 local time |
| EYUPDF04 | N/A | 07:55 to 08:00 local time |

Topology definitions for EYUPLX02

Table 12 through Table 14 on page 107 list the Starter Set topology definitions created with EYUPLX02 as the context.

Table 12. CICS system definitions for EYUPLX02

| Name | Belongs to | Description |
|----------|----------------------|--|
| EYUMAS2B | EYUCSG02 | A production CICS Transaction Server for OS/390 Release 3 TOR on system B; its hours of availability are as defined by time-period definition EYUPDF02 |
| EYUMAS3B | EYUCSG02 EYUCSG07 | A production CICS Transaction Server for OS/390 Release 3 AOR on system B; its hours of availability are as defined by time-period definition EYUPDF03 |
| EYUMAS4B | EYUCSG02 EYUCSG09 | A production CICS Transaction Server for OS/390 Release 3 FOR on system B; its hours of availability are as defined by time-period definition EYUPDF04 |

Table 13. CICS system group definitions for EYUPLX02

| Name | Belongs to | Description |
|----------|------------|-------------------------------------|
| EYUCSG02 | N/A | All TORs, AORs and FORs on system B |
| EYUCSG06 | N/A | All AORs in EYUPLX02 |
| EYUCSG07 | EYUCSG06 | System B AORs in EYUPLX02 |
| EYUCSG09 | N/A | All FORs in EYUPLX02 |

Table 14. Time-period definitions for EYUPLX02

| Name | Belongs to | Description |
|----------|------------|---------------------------|
| EYUPDF01 | N/A | 08:00 to 17:00 local time |
| EYUPDF02 | N/A | 05:00 to 01:00 local time |
| EYUPDF03 | N/A | 03:00 to 01:00 local time |
| EYUPDF04 | N/A | 07:00 to 05:00 local time |

```

27FEB2005 21:16:27 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1
W1 =SYSGRPC=====EYUPLX01=EYUPLX01=27FEB2005==21:16:27=CPSM=====10===
CMD Group  Member  Member
--- Name---- Name---- Type----
EYUCSG01  EYUMAS1B  CICSSYS
EYUCSG01  EYUMAS2A  CICSSYS
EYUCSG01  EYUMAS3A  CICSSYS
EYUCSG01  EYUMAS4A  CICSSYS
EYUCSG02  EYUMAS2A  CICSSYS
EYUCSG03  EYUCSG04  SYSGROUP
EYUCSG03  EYUCSG05  SYSGROUP
EYUCSG04  EYUMAS2A  CICSSYS
EYUCSG04  EYUMAS3A  CICSSYS
EYUCSG05  EYUMAS1B  CICSSYS

```

Figure 23. A Starter Set topology view. CICSplex SM view SYSGRPC lists the CICS systems in a specified scope. This example lists the CICS system groups in EYUPLX01, and identifies the CICS systems and CICS system groups that belong to those four groups.

BAS definitions

The Starter Set BAS definitions include resource definitions, communication links, resource groups, resource assignments, and resource descriptions.

BAS definitions for EYUPLX01

Table 15 through Table 21 on page 109 list the Starter Set BAS definitions created with EYUPLX01 as the context. Figure 24 on page 111 shows a Starter Set BAS view (RASGNDEF) that shows the characteristics of the sample resource assignments used to assign selected resource definitions to CICS systems. For each resource assignment, the resource definitions to be assigned are of a single resource type and are associated with a resource group. Figure 25 on page 111 shows the RASINDSC view that lists the resource descriptions with which the resource assignments are associated.

Table 15. Communication resource definitions for EYUPLX01

| Name | Associated with | Description |
|----------|-----------------|--|
| C002 | | Defines connection characteristics of MRO connections. |
| S0002 | C002 | Defines session characteristics of MRO connections with the connection characteristics of C002. |
| EYUMAS1A | EYUMAS2A | Defines the MRO communication link between EYUMAS1A and EYUMAS2A using the characteristics of connection definition C002 and session definition S0002. |

Table 15. Communication resource definitions for EYUPLX01 (continued)

| Name | Associated with | Description |
|-----------|-----------------|---|
| EYUEMAS1A | EYUMAS3A | Defines the MRO communication link between EYUEMAS1A and EYUMAS3A using the characteristics of connection definition C002 and session definition S0002. |
| EYUMAS4A | EYUMAS2A | Defines the MRO communication link between EYUMAS4A and EYUMAS2A using the characteristics of connection definition C002 and session definition S0002. |
| EYUMAS4A | EYUMAS3A | Defines the MRO communication link between EYUMAS4A and EYUMAS3A using the characteristics of connection definition C002 and session definition S0002. |

Table 16. Resource group and resource definitions for CPUONLY application

| Name | Belongs to | Description |
|----------|------------|--|
| EYUBAG02 | N/A | Identifies the resource group for the CPU-only application. |
| ET01 | EYUBAG02 | Transaction definition (for SSET) that calls the EYUPR001 program. |
| ET02 | EYUBAG02 | Transaction definition (for SSET) that calls the EYUPR002 program. |
| EYUPR001 | EYUBAG02 | Program definition (for SSET) called by transaction ET01. |
| EYUPR002 | EYUBAG02 | Program definition (for SSET) called by transaction ET02. |

Table 17. Resource assignment for CPUONLY application

| Name | Belongs to | Description |
|----------|------------|---|
| EYUBAA04 | N/A | Resource assignment to allocate transaction definitions from resource group EYUBAG02 as remote to target CICS system EYUMAS1A and local to related CICS systems (AORs) in CICS system group EYUCSG03. |
| EYUBAA05 | N/A | Resource assignment to allocate program definitions from resource group EYUBAG02 as local to target CICS systems (AORs) in CICS system group EYUCSG03. |

Table 18. Resource description for CPUONLY application

| Name | Belongs to | Description |
|----------|------------|--|
| EYUBAD02 | CPUONLY | Resource description that sets the logical scope for the CPU-only application. |

Table 19. Resource group and resource definitions for autoinstall programs

| Name | Belongs to | Description |
|----------|------------|---|
| EYUBAG05 | N/A | Identifies the resource group for the autoinstall programs. |

Table 19. Resource group and resource definitions for autoinstall programs (continued)

| Name | Belongs to | Description |
|----------|------------|--|
| EYUPAUTO | EYUBAG05 | Program definition for autoinstall program EYUPAUTO. |
| EYUPAUT2 | EYUBAG05 | Program definition for autoinstall program EYUPAUT2. |

Table 20. Resource assignment for autoinstall programs

| Name | Belongs to | Description |
|----------|------------|--|
| EYUBAA12 | N/A | Resource assignment to allocate program definitions from resource group EYUBAG05 as local to target CICS systems (AORs) in CICS system group EYUCSG01. |

Table 21. Resource description for autoinstall programs

| Name | Belongs to | Description |
|----------|------------|---|
| EYUBAD05 | COMMON | Resource description that sets the logical scope for the autoinstall programs and other common resources. |

BAS definitions for EYUPLX02

Table 22 through Table 24 list the Starter Set BAS definitions created with EYUPLX02 as the context.

Table 22. Resource group and resource definitions for File04 application

| Name | Belongs to | Description |
|----------|------------|--|
| EYUBAG04 | N/A | Identifies the resource group for the File04 application. |
| ET04 | EYUBAG04 | Transaction definition for ET04 transaction that calls the program EYUPR004. |
| EYUPR004 | EYUBAG04 | Program definition for EYUPR004 program called by the ET04 transaction. |
| EYUFIL04 | EYUBAG04 | File definition for EYUFIL04 file. |

Table 23. Resource assignment for File04 application

| Name | Belongs to | Description |
|----------|------------|--|
| EYUBAA09 | N/A | Resource assignment to allocate transaction definition ET04 from the resource group EYUBAG04 as remote to target CICS system EYUMAS3B. |
| EYUBAA10 | N/A | Resource assignment to allocate program definition EYUPR004 from the resource group EYUBAG04 as remote to target CICS system EYUMAS3B. |
| EYUBAA11 | EYUBAG04 | Resource assignment to allocate file definition for EYUFIL04 file from the resource group EYUBAG04 as remote to target CICS system EYUMAS3B. |

Table 24. Resource description for CPUONLY application

| Name | Belongs to | Description |
|----------|------------|---|
| EYUBAD04 | | Resource description that sets the logical scope for the CPUONLY application. |

BAS definitions for common terminals and typeterms

Table 25 through Table 26 list the Starter Set BAS definitions created with EYUPLX02 as the context.

Table 25. Resource group and resource description for terminals and typeterms

| Name | Belongs to | Description |
|----------|------------|---|
| EYUBAG06 | COMMON | Resource group for common terminal and typeterm definitions. |
| CONSL000 | EYUBAG06 | Typeterm definition for consoles. |
| MRDTYPE | EYUBAG06 | Typeterm definition for IBM Personal Computer 3270 simulator. |
| TCSN3277 | EYUBAG06 | Typeterm definition for 3277 terminal. |
| TYPEFEPI | EYUBAG06 | Typeterm definition for LUTYPE terminal used for FEPI. |
| 327R | EYUBAG06 | Typeterm definition for 3270 terminal. |
| CN00 | EYUBAG06 | Terminal definition for console CN00 associated with typeterm CONSL000. |
| CN01 | EYUBAG06 | Terminal definition for console CN01 associated with typeterm CONSL000. Note: Identical terminal definitions are provided for consoles CN02 through CN22. |
| FEPI | EYUBAG06 | Terminal definition for autoinstall model FEPI with typeterm TYPEFEPI. |
| MRD | EYUBAG06 | Terminal definition for autoinstall model MRD3270 with typeterm MRDTYPE. |
| TCSN | EYUBAG06 | Terminal definition for autoinstall model TCSN3277 with typeterm TCSN3277. |
| 3277 | EYUBAG06 | Terminal definition for autoinstall model ATRMODEL with typeterm 327R. |
| FREE | EYUBAG06 | Transaction definition for transaction used to free the keyboard of remote terminals. Associated with program FREEKB. |
| FREEKB | EYUBAG06 | Program definition for program used to free the keyboard of remote terminals. Associated with transaction FREE. |

Table 26. Resource assignment for terminals and typeterms

| Name | Belongs to | Description |
|----------|------------|--|
| EYUBAA13 | N/A | Resource assignment to allocate terminal definitions from resource group EYUBAG06 as local to target CICS systems in CICS system group EYUCSG01. |
| EYUBAA14 | N/A | Resource assignment to allocate typeterm definitions from resource group EYUBAG06 as local to target CICS systems in CICS system group EYUCSG01. |
| EYUBAA15 | N/A | Resource assignment to allocate program definitions from resource group EYUBAG06 as local to target CICS systems in CICS system group EYUCSG01. |

Table 26. Resource assignment for terminals and typeterms (continued)

| Name | Belongs to | Description |
|----------|------------|---|
| EYUBAA16 | N/A | Resource assignment to allocate transaction definitions from resource group EYUBAG06 as local to target CICS systems in CICS system group EYUCSG01. |

```

27FEB2005 19:33:51 ----- INFORMATION DISPLAY -----
COMMAND ==>                                SCROLL ==> PAGE
CURR WIN ==> 1          ALT WIN ==>
W1 =RASGNDEF=====EYUPLX01=EYUPLX01=27FEB2005==19:33:51=CPSM=====9==
CMD Name      ResGroup ResType  Target  Usage  Related  Description
-----
EYUBAA03 EYUBAG01 FILEDEF  EYUCSG03 REMOTE EYUMAS4A SSET - Allocate File
EYUBAA04 EYUBAG02 TRANDEF  EYUMAS1A REMOTE EYUCSG03 SSET - Allocate Tran
EYUBAA05 EYUBAG02 PROGDEF  EYUCSG03 LOCAL          SSET - Allocate Prog
EYUBAA12 EYUBAG05 PROGDEF  EYUCSG01 LOCAL          SSET - Autoinst Prog
EYUBAA13 EYUBAG06 TERMDEF  EYUCSG01 LOCAL
EYUBAA14 EYUBAG06 TYPTMDEF EYUCSG01 LOCAL
EYUBAA15 EYUBAG06 PROGDEF  EYUCSG01 LOCAL
EYUBAA16 EYUBAG06 TRANDEF  EYUCSG01 LOCAL
EYUBAA17 EYUBAG06 TSMDEF   EYUCSG01 LOCAL          TSMdel definition assign
  
```

Figure 24. A Starter Set BAS view. CICSplex SM view RASGNDEF shows the characteristics of the sample resource assignments used to assign selected resource definitions to CICS systems. For each resource assignment, the resource definitions to be assigned are of a single resource type and are associated with a resource group. A resource assignment must be associated with at least one resource description (RESDESC) before any assignment can begin, as shown in Figure 25 (RASINDSC view).

```

27FEB2005 19:34:15 ----- INFORMATION DISPLAY -----
COMMAND ==>                                SCROLL ==> PAGE
CURR WIN ==> 1          ALT WIN ==>
=W1 =RASINDSC=====EYUPLX01=EYUPLX01=27FEB2005==19:34:15=CPSM=====7==
CMD Name      Assign  ResGroup Target  Related  Description
-----
EYUBAD02 EYUBAA04
EYUBAD02 EYUBAA05
EYUBAD05 EYUBAA12
EYUBAD05 EYUBAA13
EYUBAD05 EYUBAA14
EYUBAD05 EYUBAA15
EYUBAD05 EYUBAA16
  
```

Figure 25. A Starter Set BAS view. CICSplex SM view RASINDSC lists the sample resource assignments and their associated resource descriptions.

Sample Workload definitions

The Starter Set Workload definitions include workload specifications, workload definitions, workload groups, and transaction groups.

Workload definitions for EYUPLX01

Table 27 on page 112 through Table 30 on page 112 list the Starter Set WLM definitions created with EYUPLX01 as the context. Figure 26 on page 113 shows a Starter Set WLM view.

Table 27. Workload specifications for EYUPLX01

| Name | Belongs to | Description |
|----------|------------|--|
| EYUWLS01 | EYUMAS1A | Establishes workload balancing in EYUPLX01 and is associated with requesting region EYUMAS1A |

Table 28. Workload definitions for EYUPLX01

| Name | Belongs to | Description |
|----------|------------|--|
| EYUWLD01 | EYUWLG01 | Workload separation for transaction group EYUTRG01; the target scope is any target region on system A |
| EYUWLD05 | EYUWLG04 | Workload separation for transaction group EYUTRG05; the target scope is any target region in CICSplex EYUPLX01 |
| EYUWLD06 | EYUWLG05 | Workload separation for transaction group EYUTRG06; the target scope is EYUMAS1B |

Table 29. Workload groups for EYUPLX01

| Name | Belongs to | Description |
|----------|------------|--|
| EYUWLG01 | EYUWLS01 | For automatic installation of workload definition EYUWLD01 |
| EYUWLG04 | EYUWLS01 | For automatic installation of workload definition EYUWLD05 |
| EYUWLG05 | EYUWLS01 | For automatic installation of workload definition EYUWLD06 |

Table 30. Transaction groups for EYUPLX01

| Name | Belongs to | Description |
|----------|------------|--|
| EYUTRG01 | EYUWLD01 | Contains transaction ETVP |
| EYUTRG05 | EYUWLD05 | Contains transactions ETD1, ETD2, and ETD3 |
| EYUTRG06 | EYUWLD06 | Contains transactions ETL1 and ETL2. ETL1 and ETL2 constitute a pseudoconversation transaction and must be processed by the same target region |

Workload definitions for EYUPLX02

Table 31 through Table 34 on page 113 list the Starter Set WLM definitions created with EYUPLX02 as the context.

Table 31. Workload specifications for EYUPLX02

| Name | Belongs to | Description |
|----------|------------|--|
| EYUWLS02 | EYUMAS2B | Establishes workload balancing in EYUPLX02 and is associated with target region EYUMAS2B |

Table 32. Workload definitions for EYUPLX02

| Name | Belongs to | Description |
|----------|------------|--|
| EYUWLD02 | EYUWLG02 | Workload separation for transaction group EYUTRG02 |
| EYUWLD03 | EYUWLG03 | Workload separation for transaction group EYUTRG03 |

Table 32. Workload definitions for EYUPLX02 (continued)

| Name | Belongs to | Description |
|----------|------------|--|
| EYUWLD04 | EYUWLG02 | Workload separation for transaction group EYUTRG04 |

Table 33. Workload groups for EYUPLX02

| Name | Belongs to | Description |
|----------|------------|--|
| EYUWLG02 | EYUWLS02 | For automatic installation of workload definition EYUWLD02 |
| EYUWLG03 | EYUWLS02 | For automatic installation of workload definition EYUWLD03 |

Table 34. Transaction groups for EYUPLX02

| Name | Belongs to | Description |
|----------|------------|--|
| EYUTRG02 | EYUWLD02 | Contains transaction ETVF |
| EYUTRG03 | EYUWLD03 | Contains transactions ETL1, ETL2, ETL3, and ETL4 |
| EYUTRG04 | EYUWLD04 | Contains transactions ETA1, ETA2, ETA3, and ETA4 |

```

27FEB2005 22:03:16 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1
>W1 =WLMSPEC=====EYUPLX01=EYUPLX01=27FEB2005==22:03:16=CPSM=====1
CMD Name      Affinity Affinity Target  Cre Match  Event      Description
---          -Relation Lifetime Scope--- Aff Key--- Name-----
EYUWLS01                                EYUCSG03  USERID      SSet Basic worklo

```

Figure 26. A Starter Set workload management view. This example shows the WLMSPEC view, which is displaying the workload specifications for CICSplex EYUPLX01.

Sample Monitor definitions

The Starter Set Monitor definitions include monitor specifications, monitor definitions, and monitor groups.

Monitor definitions for EYUPLX01

Table 35 through Table 37 on page 114 list the Starter Set MON definitions created with EYUPLX01 as the context. Figure 27 on page 115 shows a Starter Set MON view.

Table 35. Monitor specifications for EYUPLX01

| Name | Belongs to | Description |
|----------|------------|---|
| EYUMOS01 | N/A | Establishes MON for EYUMAS1A |
| EYUMOS02 | N/A | Establishes MON for CICS system group EYUCSG03 (EYUMAS2A, EYUMAS3A, and EYUMAS1B) |
| EYUMOS03 | N/A | Establishes MON for EYUMAS4A |

Table 36. Monitor definitions for EYUPLX01

| Name | Belongs to | Description |
|----------|------------|--|
| EYUMOD01 | EYUMOG01 | Monitor definition for all connections |

Table 36. Monitor definitions for EYUPLX01 (continued)

| Name | Belongs to | Description |
|----------|------------|---|
| EYUMOD02 | EYUMOG02 | Monitor definition for all transactions whose names begin "CO" |
| EYUMOD03 | EYUMOG02 | Monitor definition for all transient data queues whose names begin "CO" |
| EYUMOD04 | EYUMOG02 | Monitor definition for all transient data queues whose names begin "EQ" |
| EYUMOD05 | EYUMOG02 | Monitor definition for the DFHCSD file |
| EYUMOD06 | EYUMOG02 | Monitor definition for all journals |
| EYUMOD07 | EYUMOG03 | Monitor definition for all terminals whose lunames begin "SP" |
| EYUMOD08 | EYUMOG03 | Monitor definition for the CEMT transaction |
| EYUMOD09 | EYUMOG04 | Monitor definition for all files whose names begin "EYUF" |
| EYUMOD10 | EYUMOG05 | Monitor definition for all transactions whose names begin "ET" |
| EYUMOD11 | EYUMOG06 | Monitor definition for all programs whose names begin "EYU" |

Table 37. Monitor groups for EYUPLX01

| Name | Belongs to | Description |
|----------|----------------------------------|--|
| EYUMOG01 | EYUMOS01 EYUMOS02 EYUMOS03 | For automatic installation of monitor definition EYUMOD01 |
| EYUMOG02 | EYUMOS01 EYUMOS02 EYUMOS03 | For automatic installation of monitor definitions EYUMOD02, EYUMOD03, EYUMOD04, and EYUMOD05, and EYUMOD06 |
| EYUMOG03 | EYUMOS01 | For automatic installation of monitor definitions EYUMOD07 and EYUMOD08 |
| EYUMOG04 | EYUMOS01 EYUMOS02 EYUMOS03 | For automatic installation of monitor definition EYUMOD09 |
| EYUMOG05 | EYUMOS01 EYUMOS02 | For automatic installation of monitor definition EYUMOD10 |
| EYUMOG06 | EYUMOS02 | For automatic installation of monitor definition EYUMOD11 |

Monitor definitions for EYUPLX02

Table 38 through Table 40 on page 115 list the Starter Set MON definitions created with EYUPLX02 as the context.

Table 38. Monitor specifications for EYUPLX02

| Name | Belongs to | Description |
|----------|----------------------|------------------------------|
| EYUMOS01 | EYUMAS2B | Establishes MON for EYUMAS2B |
| EYUMOS02 | EYUCSG07 EYUCSG08 | Establishes MON for EYUMAS3B |
| EYUMOS03 | EYUMAS4B | Establishes MON for EYUMAS4B |

Table 39. Monitor definitions for EYUPLX02

| Name | Belongs to | Description |
|----------|------------|--|
| EYUMOD01 | EYUMOG01 | Monitor definition for all connections, except the connection to the remote MAS. |
| EYUMOD02 | EYUMOG01 | Monitor definition for all transient data queues |
| EYUMOD03 | EYUMOG01 | Monitor definition for all journals |
| EYUMOD04 | EYUMOG05 | Monitor definition for the connection to the remote MAS |
| EYUMOD05 | EYUMOG02 | Monitor definition for all transactions whose names begin "ET" |
| EYUMOD06 | EYUMOG03 | Monitor definition for all programs whose names begin "EYUPR" |
| EYUMOD07 | EYUMOG04 | Monitor definition for all files whose names begin "EYUF" |

Table 40. Monitor groups for EYUPLX02

| Name | Belongs to | Description |
|----------|----------------------------------|--|
| EYUMOG01 | EYUMOS01 EYUMOS02 EYUMOS03 | For automatic installation of monitor definitions EYUMOD01, EYUMOD02, and EYUMOD03 |
| EYUMOG02 | EYUMOS01 EYUMOS02 | For automatic installation of monitor definition EUMOD05 |
| EYUMOG03 | EYUMOS02 | For automatic installation of monitor definition EYUMOD06 |
| EYUMOG04 | EYUMOS02 EYUMOS03 | For automatic installation of monitor definition EYUMOD07 |
| EYUMOG05 | EYUMOS01 | For automatic installation of monitor definition EYUMOD04 |

```

27FEB2005 19:33:23 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1
W1 =MONDEF=====EYUPLX01=EYUPLX01=27FEB2005==19:33:23=CPSM=====11===
CMD Def Resource Resource Include Res Monitor Definition
--- Name---- Name---- Type--- Data--- Stat Description-----
EYUMOD01 * MCONN YES YES SSet - All Connections
EYUMOD02 CO* MTRAN YES NO SSet - All CO* Transactions
EYUMOD03 CO* MTDQS YES NO SSet - TDQs CO*
EYUMOD04 EQ* MTDQS NO YES SSet - TDQs EQ*
EYUMOD05 DFHCSD MFILE YES YES SSet - DFHCSD Files
EYUMOD06 * MJRNL YES YES SSet - All Journals
EYUMOD07 SP* MTERM YES YES SSet - All SP* Terminals
EYUMOD08 CEMT MTRAN YES YES SSet - CEMT Transaction
EYUMOD09 EYUF* MFILE YES YES SSet - EYUF* files
EYUMOD10 ET* MTRAN YES YES SSet - ET* Transations
EYUMOD11 EYU* MPROG YES YES SSet - EYU* Programs
  
```

Figure 27. A Starter Set monitoring view. The MONDEF view displays all monitor definitions for CICSplex EYUPLX01.

Sample Analysis definitions

The Starter Set analysis definitions include: analysis point specifications; analysis specifications; analysis groups; analysis definitions; evaluation definitions; action definitions; and status definitions.

Analysis definitions for EYUPLX01

Table 41 through Table 46 on page 117 list the Starter Set RTA definitions created with EYUPLX01 as the context. Figure 28 on page 118 shows a Starter Set RTA view.

Table 41. Analysis point specifications for EYUPLX01

| Name | Belongs to | Description |
|----------|------------|--|
| EYURAP01 | EYUPLX01 | EYUCMS1A is the CMAS responsible for the analysis definitions in analysis group EYURTG02 within EYUPLX01 |
| EYURAPLC | EYUPLX01 | EYUCMS1A is the CMAS responsible for the analysis definitions in analysis group EYURTGLC within EYUPLX01 |

Table 42. Analysis specifications for EYUPLX01

| Name | Belongs to | Description |
|----------|------------|------------------|
| EYURTS01 | EYUMAS1A | For TOR EYUMAS1A |
| EYURTS03 | EYUMAS4A | For FOR EYUMAS4A |

Table 43. Analysis groups for EYUPLX01

| Name | Belongs to | Description |
|----------|------------|---|
| EYURTG01 | EYURTS03 | For automatic installation of analysis definitions EYURTD16 and EYURTD17 |
| EYURTG02 | EYURAP01 | For automatic installation of analysis definitions EYURTD01, EYURTD02, EYURTD19, EYURTD20, EYURTD21, and EYURTD22 |
| EYURTG07 | EYURTS01 | For automatic installation of analysis definition EYURTD18 |

Table 44. Analysis definitions for EYUPLX01

| Name | Belongs to | Description |
|----------|------------|---|
| EYURTD01 | EYURTG02 | Evaluation definition EYURTE01 activates action definition EYURTA01 |
| EYURTD02 | EYURTG02 | Evaluation definition EYURTE02 activates action definition EYURTA02 |
| EYURTD16 | EYURTG01 | Evaluation definition EYURTE16 activates action definition EYURTA16 |
| EYURTD17 | EYURTG01 | Evaluation definition EYURTE17 activates action definition EYURTA17 |
| EYURTD18 | EYURTG07 | Evaluation definition EYURTE18 activates action definition EYURTA18 |
| EYURTD19 | EYURTG02 | Evaluation definition EYURTE19 activates action definition EYURTA19 |

Table 44. Analysis definitions for EYUPLX01 (continued)

| Name | Belongs to | Description |
|----------|------------|---|
| EYURTD20 | EYURTG02 | Evaluation definition EYURTE20 activates action definition EYURTA20 |
| EYURTD21 | EYURTG02 | Evaluation definition EYURTE21 activates action definition EYURTA21 |
| EYURTD22 | EYURTG02 | Evaluation definition EYURTE22 activates action definition EYURTA22 |

Table 45. Evaluation definitions for EYUPLX01

| Name | Belongs to | Description |
|----------|------------|---|
| EYURTE01 | EYURTD01 | Notify when any connection is not ACQUIRED |
| EYURTE02 | EYURTD02 | Notify when any DFHCSD file is not ENABLED |
| EYURTE16 | EYURTD16 | Notify when any file requires Emptying |
| EYURTE17 | EYURTD17 | Notify when the maximum number of DBCTL threads is active |
| EYURTE18 | EYURTD18 | Notify when the dynamic transaction routing program is not EYU9XLOP |
| EYURTE19 | EYURTD19 | Notify when monitoring is not active in a CICS system |
| EYURTE20 | EYURTD20 | Notify when the percentage of free space in the DSA drops to a specified threshold |
| EYURTE21 | EYURTD21 | Notify when the number of outstanding event notifications reaches a specified threshold |
| EYURTE22 | EYURTD22 | Notify when TRANDUMP dump suppression reaches a specified threshold |

Table 46. Action definitions for EYUPLX01

| Name | Belongs to | Description |
|----------|------------|--|
| EYURTA01 | EYURTD01 | Generate event notification and external message when a connection is unavailable |
| EYURTA02 | EYURTD02 | Generate event notification and external message when a file is unavailable |
| EYURTA16 | EYURTD16 | Generate event notification and external message when a file requires Emptying |
| EYURTA17 | EYURTD17 | Generate event notification and external message when the maximum number of DBCTL threads is active |
| EYURTA18 | EYURTD18 | Generate event notification and external message when the dynamic transaction routing program is not EYU9XLOP |
| EYURTA19 | EYURTD19 | Generate event notification and external message when monitoring is not active in a CICS system |
| EYURTA20 | EYURTD20 | Generate event notification and external message when the percentage of free space in the DSA drops to a specified threshold |
| EYURTA21 | EYURTD21 | Generate event notification and external message when a specified number of outstanding event notifications is reached |

Table 46. Action definitions for EYUPLX01 (continued)

| Name | Belongs to | Description |
|----------|------------|---|
| EYURTA22 | EYURTD22 | Generate event notification and external message when a specified number of dumps is being suppressed by TRANDUMP suppression |

```

27FEB2005 21:02:56 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1
W1 =RTAGROUP=====EYUPLX01=EYUPLX01=27FEB2005==21:02:56=CPSM=====4===
CMD Analysis Description
--- Group--- -----
EYURGLC SSet - License
EYURTG01 SSet - FOR related definitions
EYURTG02 SSet - Plex wide definitions
EYURTG07 SSet - TOR related definitions
    
```

Figure 28. A Starter Set RTA view. This example shows the RTAGROUP view, which is displaying the analysis groups in EYUPLX01.

Analysis definitions for EYUPLX02

Table 47 through Table 54 on page 124 list the Starter Set RTA definitions created with EYUPLX02 as the context.

Table 47. Analysis point specifications for EYUPLX02

| Name | Belongs to | Description |
|----------|------------|--|
| EYURAP02 | EYUPLX02 | EYUCMS1B is the CMAS responsible for the analysis definitions in analysis group EYURTG04 within EYUPLX02 |
| EYURAP03 | EYUCSG07 | EYUCMS1B is the CMAS responsible for the analysis definitions in analysis group EYURTG05 within CICS system group EYUCSG07 |
| EYURAP04 | EYUCSG08 | EYUCMS1B is the CMAS responsible for the analysis definitions in analysis group EYURTG05 within CICS system group EYUCSG08 |

Table 48. Primary CMAS associations for EYUPLX02

| Name | Belongs to | Description |
|------|------------|--|
| N/A | N/A | RTA primary CMAS association between EYURAP02 and EYUCMS1B |
| N/A | N/A | RTA primary CMAS association between EYURAP03 and EYUCMS1B |
| N/A | N/A | RTA primary CMAS association between EYURAP04 and EYUCMS1B |

Table 49. Analysis specifications for EYUPLX02

| Name | Belongs to | Description |
|----------|------------|------------------|
| EYURTS02 | EYUMAS2B | For TOR EYUMAS2B |

Table 50. Analysis groups for EYUPLX02

| Name | Belongs to | Description |
|----------|----------------------|---|
| EYURTG03 | EYURTS02 | For automatic installation of analysis definition EYURTD03 during period defined by EYUPDF01 |
| EYURTG04 | EYURAP02 | For automatic installation of analysis definitions EYURTD04 (during period defined by EYUPDF01), EYURTD10, EYURTD23, EYURTD24, EYURTD25 |
| EYURTG05 | EYURAP03 EYURAP04 | For automatic installation of analysis definition EYURTD09 |
| EYURTG06 | EYURTS02 | For automatic installation of analysis definition EYURTD05 during period defined by EYUPDF01 |
| EYURTG08 | EYURTS02 | For automatic installation of status definition EYURST03 |
| EYURTG09 | EYURTS02 | For automatic installation of status definition EYURST04 |
| EYURTG10 | EYURTS02 | For automatic installation of status definition EYURST06 |

Table 51. Analysis definitions for EYUPLX02

| Name | Belongs to | Description |
|----------|------------|--|
| EYURTD03 | EYURTG03 | Evaluation definition EYURTE03 activates action definition EYURTA03 |
| EYURTD04 | EYURTG04 | Evaluation definition EYURTE04 activates action definition EYURTA04 |
| EYURTD05 | EYURTG06 | Evaluation definitions EYURTE05 and EYURTE06 (combined) activate action definition EYURTA05 |
| EYURTD06 | N/A | Status definition EYURST03 or EYURST04 activates action definition EYURTA06 |
| EYURTD08 | N/A | Status definition EYURST06 activates action definition EYURTA08 |
| EYURTD09 | EYURTG05 | Evaluation definitions EYURTE10, EYURTE11, EYURTE12, EYURTE13, or EYURTE14 activate action definition EYURTA09 |
| EYURTD10 | EYURTG04 | Evaluation definition EYURTE15 activates action definition EYURTA10 |
| EYURTD23 | EYURTG04 | Evaluation definition EYURTE23 activates action definition EYURTA23 |
| EYURTD24 | EYURTG04 | Evaluation definition EYURTE24 activates action definition EYURTA24 |
| EYURTD25 | EYURTG04 | Evaluation definition EYURTE25 activates action definition EYURTA25 |
| EYURTD26 | N/A | Evaluation definition EYURTE26 activates action definition EYURTA26 |
| EYURTD27 | N/A | Evaluation definition EYURTE27 activates action definition EYURTA27 |
| EYURTD28 | N/A | Evaluation definition EYURTE28 activates action definition EYURTA28 |
| EYURTD29 | N/A | Evaluation definition EYURTE29 activates action definition EYURTA29 |

Table 51. Analysis definitions for EYUPLX02 (continued)

| Name | Belongs to | Description |
|----------|------------|--|
| EYURTD30 | N/A | Evaluation definition EYURTE30 activates action definition EYURTA30 |
| EYURTD31 | N/A | Evaluation definition EYURTE31 activates action definition EYURTA31 |
| EYURTD32 | N/A | Evaluation definition EYURTE32 activates action definition EYURTA32 |
| EYURTD33 | N/A | Evaluation definition EYURTE33 activates action definition EYURTA33 |
| EYURTD34 | N/A | Evaluation definition EYURTE34 activates action definition EYURTA34 |
| EYURTD35 | N/A | Evaluation definition EYURTE35 activates action definition EYURTA35 |
| EYURTD36 | N/A | Evaluation definition EYURTE36 activates action definition EYURTA36 |
| EYURTD37 | N/A | Evaluation definition EYURTE37 activates action definition EYURTA37 |
| EYURTD38 | N/A | Evaluation definition EYURTE38 activates action definition EYURTA38 |
| EYURTD39 | N/A | Evaluation definition EYURTE39 activates action definition EYURTA39 |
| EYURTD40 | N/A | Evaluation definition EYURTE40 activates action definition EYURTA40 |
| EYURTD41 | N/A | Evaluation definition EYURTE41 activates action definition EYURTA41 |
| EYURTD42 | N/A | Evaluation definition EYURTE42 activates action definition EYURTA42 |
| EYURTD43 | N/A | Evaluation definition EYURTE43 activates action definition EYURTA43 |
| EYURTD44 | N/A | Evaluation definition EYURTE44 activates action definition EYURTA44 |
| EYURTD45 | N/A | Evaluation definition EYURTE45 activates action definition EYURTA45 |
| EYURTD46 | N/A | Evaluation definition EYURTE46 activates action definition EYURTA46 |
| EYURTD47 | N/A | Evaluation definition EYURTE47 activates action definition EYURTA47 |
| EYURTD48 | N/A | Evaluation definition EYURTE48 activates action definition EYURTA48 |
| EYURTD49 | N/A | Evaluation definition EYURTE49 activates action definition EYURTA49 |
| EYURTD50 | N/A | Evaluation definition EYURTE50 activates action definition EYURTA50 |
| EYURTD52 | N/A | Evaluation definitions EYURTE51 and EYURTE52 activate action definition EYURTA52 |
| EYURTD53 | N/A | Evaluation definition EYURTE53 activates action definition EYURTA53 |

Table 52. Evaluation definitions for EYUPLX02

| Name | Belongs to | Description |
|----------|------------|--|
| EYURTE03 | EYURTD03 | Trigger when the connection to a remote MAS is unavailable |
| EYURTE04 | EYURTD04 | Trigger when the response time of ET* transactions is > 2 seconds |
| EYURTE05 | EYURTD05 | Trigger when primary connection is not available |
| EYURTE06 | EYURTD05 | Trigger when secondary connection is not available |
| EYURTE10 | EYURTD09 | Trigger when programs in an application are not ENABLED |
| EYURTE11 | EYURTD09 | Trigger when ENABLED files in an application are not OPEN |
| EYURTE12 | EYURTD09 | Trigger when files in an application are not ENABLED |
| EYURTE13 | EYURTD09 | Trigger when a DB2 database in an application is not ACTIVE |
| EYURTE14 | EYURTD09 | Trigger when local transactions in an application are not ENABLED |
| EYURTE15 | EYURTD10 | Trigger when the number of priority 1 events reaches specified thresholds |
| EYURTE23 | EYURTD23 | Trigger when a remote Q output rate grows |
| EYURTE24 | EYURTD24 | Trigger when special terminals become unavailable |
| EYURTE25 | EYURTD25 | Trigger when a program use count exceeds specified levels |
| EYURTE26 | EYURTD26 | Trigger when CICS CPU usage reaches a specified threshold |
| EYURTE27 | EYURTD27 | Trigger when the largest DSA free size reduces to a specified threshold |
| EYURTE28 | EYURTD28 | Trigger when the number of transaction dumps reach a specified threshold |
| EYURTE29 | EYURTD29 | Trigger when the number of active LSRPOOL strings reaches a specified threshold |
| EYURTE30 | EYURTD30 | Trigger when the number of LSRPOOL waits reaches a specified threshold |
| EYURTE31 | EYURTD31 | Trigger when the number of current file string waits reaches a specified threshold |
| EYURTE32 | EYURTD32 | Trigger when the number of storage violations for all transactions reaches a specified threshold |
| EYURTE33 | EYURTD33 | Trigger when the number of storage violations for all terminals reaches a specified threshold |
| EYURTE34 | EYURTD34 | Trigger when the number of items on a transient data queue reaches a specified threshold |
| EYURTE35 | EYURTD35 | Trigger when the number of current transient data strings reaches a specified threshold |
| EYURTE36 | EYURTD36 | Trigger when the number of transient data string waits reaches a specified threshold |
| EYURTE37 | EYURTD37 | Trigger when the number of auxiliary temporary storage queue items reaches a specified threshold |

Table 52. Evaluation definitions for EYUPLX02 (continued)

| Name | Belongs to | Description |
|----------|------------|---|
| EYURTE38 | EYURTD38 | Trigger when the number of current temporary storage strings reaches a specified threshold |
| EYURTE39 | EYURTD39 | Trigger when the number of temporary storage string waits reaches a specified threshold |
| EYURTE40 | EYURTD40 | Trigger when the number of storage violations for a transaction reaches a specified threshold |
| EYURTE41 | EYURTD41 | Trigger when the number of storage violations for a terminal reaches a specified threshold |
| EYURTE42 | EYURTD42 | Trigger when the page-in rate for CICS reaches a specified threshold |
| EYURTE43 | EYURTD43 | Trigger when the number of MRO links in use reaches a specified threshold |
| EYURTE44 | EYURTD44 | Trigger when the number of ISC links in use reaches a specified threshold |
| EYURTE45 | EYURTD45 | Trigger when the total of system dumps reaches a specified threshold |
| EYURTE46 | EYURTD46 | Trigger when use of the DSA above the 16M line reaches a specified threshold |
| EYURTE47 | EYURTD47 | Trigger when use of the DSA below the 16M line reaches a specified threshold |
| EYURTE48 | EYURTD48 | Trigger when the number of I/O requests for a CICS system reaches a specified threshold |
| EYURTE49 | EYURTD49 | Trigger when the DBCTL attachment interface is inactive |
| EYURTE50 | EYURTD50 | Trigger when the DB2 attachment interface is inactive |
| EYURTE51 | EYURTD51 | Trigger when the DB2 attachment interface is active (for use with CICS shutdown) |
| EYURTE52 | EYURTD52 | Trigger when CICS is in shutdown stage |
| EYURTE53 | EYURTD53 | Trigger when the number of DB2 threads in use reaches a specified threshold |

Table 53. Action definitions for EYUPLX02

| Name | Belongs to | Description |
|----------|------------|--|
| EYURTA03 | EYURTD03 | Generate event notification and external message to warn that a remote MAS connection is unavailable |
| EYURTA04 | EYURTD04 | Generate event notification and external message to indicate the response time is too high for ET* transactions |
| EYURTA05 | EYURTD05 | Generate event notification and external message to indicate primary and secondary connections to a region are unavailable |
| EYURTA06 | EYURTD06 | Generate event notification and external message to indicate a Client/Server connection is unavailable |
| EYURTA08 | EYURTD08 | Generate event notification and external message to indicate that DB2 is not correctly initialized |
| EYURTA09 | EYURTD09 | Generate event notification and external message to indicate a total application is unavailable |

Table 53. Action definitions for EYUPLX02 (continued)

| Name | Belongs to | Description |
|----------|------------|---|
| EYURTA10 | EYURTD10 | Generate event notification and external message to indicate the number of priority 1 events is high |
| EYURTA23 | EYURTD23 | Generate event notification and external message to indicate a remote queue output level is high |
| EYURTA24 | EYURTD24 | Generate event notification and external message to indicate specially designated terminals are unavailable |
| EYURTA25 | EYURTD25 | Generate event notification and external message to indicate a program use count is high |
| EYURTA26 | EYURTD26 | Generate event notification and external message when CICS CPU usage reaches a specified threshold |
| EYURTA27 | EYURTD27 | Generate event notification and external message when the largest DSA free size reduces to a specified threshold |
| EYURTA28 | EYURTD28 | Generate event notification and external message when the number of transaction dumps reaches a specified threshold |
| EYURTA29 | EYURTD29 | Generate event notification and external message when the number of active LSRPOOL strings reaches a specified threshold |
| EYURTA30 | EYURTD30 | Generate event notification and external message when the number of LSRPOOL waits reaches a specified threshold |
| EYURTA31 | EYURTD31 | Generate event notification and external message when the number of current file string waits reaches a specified threshold |
| EYURTA32 | EYURTD32 | Generate event notification and external message when the number of storage violations for all transactions reaches a specified threshold |
| EYURTA33 | EYURTD33 | Generate event notification and external message when the number of storage violations for all terminals reaches a specified threshold |
| EYURTA34 | EYURTD34 | Generate event notification and external message when the number of items on a transient data queue reaches a specified threshold |
| EYURTA35 | EYURTD35 | Generate event notification and external message when the number of current transient data strings reaches a specified threshold |
| EYURTA36 | EYURTD36 | Generate event notification and external message when the number of transient data string waits reaches a specified threshold |
| EYURTA37 | EYURTD37 | Generate event notification and external message when the number of auxiliary temporary storage queue items reaches a specified threshold |
| EYURTA38 | EYURTD38 | Generate event notification and external message when the number of active temporary storage strings reaches a specified threshold |
| EYURTA39 | EYURTD39 | Generate event notification and external message when the number of temporary storage string waits reaches a specified threshold |

Table 53. Action definitions for EYUPLX02 (continued)

| Name | Belongs to | Description |
|----------|------------|--|
| EYURTA40 | EYURTD40 | Generate event notification and external message when the number of storage violations for a transaction reaches a specified threshold |
| EYURTA41 | EYURTD41 | Generate event notification and external message when the number of storage violations for a terminal reaches a specified threshold |
| EYURTA42 | EYURTD42 | Generate event notification and external message when the page-in rate for CICS reaches a specified threshold |
| EYURTA43 | EYURTD43 | Generate event notification and external message when the number of MRO links in use reaches a specified threshold |
| EYURTA44 | EYURTD44 | Generate event notification and external message when the number of ISC links in use reaches a specified threshold |
| EYURTA45 | EYURTD45 | Generate event notification and external message when the total of system dumps reaches a specified threshold |
| EYURTA46 | EYURTD46 | Generate event notification and external message when use of the DSA above the 16M line reaches a specified threshold |
| EYURTA47 | EYURTD47 | Generate event notification and external message when use of the DSA below the 16M line reaches a specified threshold |
| EYURTA48 | EYURTD48 | Generate event notification and external message when the number of I/O requests for a CICS system reaches a specified threshold |
| EYURTA49 | EYURTD49 | Generate event notification and external message when the DBCTL attachment interface is inactive |
| EYURTA50 | EYURTD50 | Generate event notification and external message when the DB2 attachment interface is inactive |
| EYURTA52 | EYURTD52 | Generate event notification and external message when CICS is in shutdown stage |
| EYURTA53 | EYURTD53 | Generate event notification and external message when the number of DB2 threads in use reaches a specified threshold |
| EYURTA6K | EYURTD6K | Generate event notification and external message when a client/server probe has been triggered |
| EYURTA4P | EYURTD4P | Generate event notification and external message when a CICS/400 probe has been triggered |
| EYURTADB | EYURTDDB | Generate event notification and external message when a DB2 Application status probe has been triggered |

Table 54. Status definitions for EYUPLX02

| Name | Belongs to | Description |
|----------|------------|---|
| EYURST03 | EYURTG08 | Trigger when Client/Server probe returns true token |
| EYURST04 | EYURTG09 | Trigger when CICS/400 probe returns true token |

Table 54. Status definitions for EYUPLX02 (continued)

| Name | Belongs to | Description |
|----------|------------|--|
| EYURST06 | EYURTG10 | Trigger when DB2 Application status probe returns true token |

The status definitions supplied with the Starter Set are triggered by CICS programs known as *status probes*. These programs are shipped in source format in data set CICSTS31.CPSM.SEYUDEF. You are recommended to read the commented source files before using any of these programs. The status probe programs and the definitions that use them are:

| Status Probe | Status Definition | Environment |
|--------------|-------------------|---------------------|
| EYUPRB60 | EYURST03 | IBM CICS for AIX® |
| EYUPRB6B | EYURST03 | IBM CICS for AIX |
| EYUPRB40 | EYURST04 | IBM CICS for OS/400 |
| EYUPRB4B | EYURST04 | IBM CICS for OS/400 |
| EYUPRBDB | EYURST06 | DB2 |

Two sample JCL members have been added to data set CICSTS31.CPSM.SEYUJCL to create the RTA status probe load modules: member EYUJPRDA creates the load module for status probe EYUPRBDB; member EYUJPRBA creates the load modules for all status probes except EYUPRBDB.

The Starter Set naming convention

All of the supplied Starter Set components follow a naming convention that allows you both to identify the contents of a particular dataset member and to differentiate the Starter Set from “live” CICSplex SM data.

The 8-character names of all Starter Set components comprise:

- A 3-character product symbol, “EYU”.
- A 3-character resource-type identifier. For example, the characters “RTS” identify an analysis specification, and the characters “PDF” identify a time-period definition.
- A 2-character, site-specific indicator. Typically, this identifies the system on which the component is installed. For example, EYUCAS1A is CAS number 1 on System A, and EYUMAS4B is MAS number 4 on System B.

Thus, the Starter Set naming convention is as follows:

Naming convention
CICSplex SM resource type
EYUBAAxx
resource assignment
EYUBADxx
resource description
EYUBAGxx
resource group
EYUCASxx
CAS
EYUCMSxx
CMAS

EYUCSGxx
CICS system group

EYUMASxx
MAS

EYUMODxx
Monitor definition

EYUMOGxx
Monitor group

EYUMOSxx
Monitor specification

EYUPDFxx
Time-period definition

EYUPLXxx
CICSplex

EYURAPxx
Analysis point specification

EYURSTxx
Status definition

EYURTAxx
Action definition

EYURTDxx
Analysis definition

EYURTExx
Evaluation definition

EYURTGxx
Analysis group

EYURTSxx
Analysis specification

EYUTRGxx
Transaction group

EYUWLDxx
Workload definition

EYUWLGxx
Workload group

EYUWLSxx
Workload specification

It is most important that you define an appropriate naming convention for your own CICSplex SM environment. How to approach this is described in Chapter 9, “Planning CICSplex SM installation and setup,” on page 91.

Bibliography

The CICS Transaction Server for z/OS library

The published information for CICS Transaction Server for z/OS is delivered in the following forms:

The CICS Transaction Server for z/OS Information Center

The CICS Transaction Server for z/OS Information Center is the primary source of user information for CICS Transaction Server. The Information Center contains:

- Information for CICS Transaction Server in HTML format.
- Licensed and unlicensed CICS Transaction Server books provided as Adobe Portable Document Format (PDF) files. You can use these files to print hardcopy of the books. For more information, see “PDF-only books.”
- Information for related products in HTML format and PDF files.

One copy of the CICS Information Center, on a CD-ROM, is provided automatically with the product. Further copies can be ordered, at no additional charge, by specifying the Information Center feature number, 7014.

Licensed documentation is available only to licensees of the product. A version of the Information Center that contains only unlicensed information is available through the publications ordering system, order number SK3T-6945.

Entitlement hardcopy books

The following essential publications, in hardcopy form, are provided automatically with the product. For more information, see “The entitlement set.”

The entitlement set

The entitlement set comprises the following hardcopy books, which are provided automatically when you order CICS Transaction Server for z/OS, Version 3 Release 1:

Memo to Licensees, GI10-2559
CICS Transaction Server for z/OS Program Directory, GI10-2586
CICS Transaction Server for z/OS Release Guide, GC34-6421
CICS Transaction Server for z/OS Installation Guide, GC34-6426
CICS Transaction Server for z/OS Licensed Program Specification, GC34-6608

You can order further copies of the following books in the entitlement set, using the order number quoted above:

CICS Transaction Server for z/OS Release Guide
CICS Transaction Server for z/OS Installation Guide
CICS Transaction Server for z/OS Licensed Program Specification

PDF-only books

The following books are available in the CICS Information Center as Adobe Portable Document Format (PDF) files:

CICS books for CICS Transaction Server for z/OS

General

CICS Transaction Server for z/OS Program Directory, GI10-2586
CICS Transaction Server for z/OS Release Guide, GC34-6421
CICS Transaction Server for z/OS Migration from CICS TS Version 2.3, GC34-6425

CICS Transaction Server for z/OS Migration from CICS TS Version 1.3,
GC34-6423

CICS Transaction Server for z/OS Migration from CICS TS Version 2.2,
GC34-6424

CICS Transaction Server for z/OS Installation Guide, GC34-6426

Administration

CICS System Definition Guide, SC34-6428

CICS Customization Guide, SC34-6429

CICS Resource Definition Guide, SC34-6430

CICS Operations and Utilities Guide, SC34-6431

CICS Supplied Transactions, SC34-6432

Programming

CICS Application Programming Guide, SC34-6433

CICS Application Programming Reference, SC34-6434

CICS System Programming Reference, SC34-6435

CICS Front End Programming Interface User's Guide, SC34-6436

CICS C++ OO Class Libraries, SC34-6437

CICS Distributed Transaction Programming Guide, SC34-6438

CICS Business Transaction Services, SC34-6439

Java Applications in CICS, SC34-6440

JCICS Class Reference, SC34-6001

Diagnosis

CICS Problem Determination Guide, SC34-6441

CICS Messages and Codes, GC34-6442

CICS Diagnosis Reference, GC34-6899

CICS Data Areas, GC34-6902

CICS Trace Entries, SC34-6443

CICS Supplementary Data Areas, GC34-6905

Communication

CICS Intercommunication Guide, SC34-6448

CICS External Interfaces Guide, SC34-6449

CICS Internet Guide, SC34-6450

Special topics

CICS Recovery and Restart Guide, SC34-6451

CICS Performance Guide, SC34-6452

CICS IMS Database Control Guide, SC34-6453

CICS RACF Security Guide, SC34-6454

CICS Shared Data Tables Guide, SC34-6455

CICS DB2 Guide, SC34-6457

CICS Debugging Tools Interfaces Reference, GC34-6908

CICSplex SM books for CICS Transaction Server for z/OS

General

CICSplex SM Concepts and Planning, SC34-6459

CICSplex SM User Interface Guide, SC34-6460

CICSplex SM Web User Interface Guide, SC34-6461

Administration and Management

CICSplex SM Administration, SC34-6462

CICSplex SM Operations Views Reference, SC34-6463

CICSplex SM Monitor Views Reference, SC34-6464

CICSplex SM Managing Workloads, SC34-6465

CICSplex SM Managing Resource Usage, SC34-6466

CICSplex SM Managing Business Applications, SC34-6467

Programming

CICSplex SM Application Programming Guide, SC34-6468

CICSplex SM Application Programming Reference, SC34-6469

Diagnosis

CICSplex SM Resource Tables Reference, SC34-6470
CICSplex SM Messages and Codes, GC34-6471
CICSplex SM Problem Determination, GC34-6472

CICS family books

Communication

CICS Family: Interproduct Communication, SC34-6473
CICS Family: Communicating from CICS on System/390, SC34-6474

Licensed publications

The following licensed publications are not included in the unlicensed version of the Information Center:

CICS Diagnosis Reference, GC34-6899
CICS Data Areas, GC34-6902
CICS Supplementary Data Areas, GC34-6905
CICS Debugging Tools Interfaces Reference, GC34-6908

Other CICS books

The following publications contain further information about CICS, but are not provided as part of CICS Transaction Server for z/OS, Version 3 Release 1.

| | |
|---|-----------|
| <i>Designing and Programming CICS Applications</i> | SR23-9692 |
| <i>CICS Application Migration Aid Guide</i> | SC33-0768 |
| <i>CICS Family: API Structure</i> | SC33-1007 |
| <i>CICS Family: Client/Server Programming</i> | SC33-1435 |
| <i>CICS Transaction Gateway for z/OS Administration</i> | SC34-5528 |
| <i>CICS Family: General Information</i> | GC33-0155 |
| <i>CICS 4.1 Sample Applications Guide</i> | SC33-1173 |
| <i>CICS/ESA 3.3 XRF Guide</i> | SC33-0661 |

Determining if a publication is current

IBM regularly updates its publications with new and changed information. When first published, both hardcopy and BookManager® softcopy versions of a publication are usually in step. However, due to the time required to print and distribute hardcopy books, the BookManager version is more likely to have had last-minute changes made to it before publication.

Subsequent updates will probably be available in softcopy before they are available in hardcopy. This means that at any time from the availability of a release, softcopy versions should be regarded as the most up-to-date.

For CICS Transaction Server books, these softcopy updates appear regularly on the *Transaction Processing and Data Collection Kit* CD-ROM, SK2T-0730-xx. Each reissue of the collection kit is indicated by an updated order number suffix (the -xx part). For example, collection kit SK2T-0730-06 is more up-to-date than SK2T-0730-05. The collection kit is also clearly dated on the cover.

Updates to the softcopy are clearly marked by revision codes (usually a # character) to the left of the changes.

Accessibility

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully.

You can perform most tasks required to set up, run, and maintain your CICSplex SM system in one of these ways:

- using a 3270 emulator connected to CICSplex SM
- using a 3270 emulator logged on to CICS
- using a 3270 emulator logged on to TSO
- using a 3270 emulator as an MVS system console
- using the CICSplex SM web user interface.

IBM Personal Communications (Version 5.0.1 for Windows 95, Windows 98, Windows NT and Windows 2000; version 4.3 for OS/2) provides 3270 emulation with accessibility features for people with disabilities. You can use this product to provide the accessibility features you need in your CICSplex SM system.

Index

A

- abend compensation 62
- ACTDEF view 78
- action command 3, 91
- ADMCONFIG view 36, 37
- administering CAS-to-CAS links 35
- administering the CMAS configuration 36
- administering the topology of a CICSplex 35
- ADMTOPOL view 38
- affinity, intertransaction 61, 64
- agent code, MAS 10
- analysis definition 78
 - Starter Set 116
- analysis group 78
- analysis point monitoring (APM)
 - automation and 73
 - description of 77
 - implementing 81
- analysis specification 78
- AOC CICS Automation 74
- APCMAS view 78
- application programming interface (API) 5
- APSPEC view 78
- automatic installation
 - of a monitor definition 88
 - of a workload definition 71
 - of an analysis definition 81
 - of resources 52

B

- batched repository-update facility 5
 - description of 5, 15, 97
 - EXTRACT routine 15, 54
 - Starter Set use of 101
- BATCHREP view 36, 97
- Business Application Services (BAS)
 - creating resource definitions 49
 - definition of 3
 - description of 41
 - EXTRACT routine 54
 - functions 41
 - objects 14
 - resource installation 41
 - resource versions 42
 - security, special considerations 94
 - Starter Set samples 107

C

- CAS-to-CAS links 35
- CASACT view 35
- CASDEF view 35
- CASDEFD view 35
- CICS business transaction services (BTS) 4
- CICS Monitoring Facility (CMF) 83, 84, 89
- CICS system 17

- CICS system group 9, 22
 - defining 38
 - identifying 38
 - Starter Set examples 102
 - using instead of CICSplexes 21
- CICS system objects 17
- CICSGRP view 39
- CICSplex
 - definition of 9
 - definitions 37
 - identifying 19
 - when to have more than one 20
- CICSplex SM address space (CMAS)
 - release level 11
- CICSplex topology
 - administering 35
 - defining 35
- CICSPLEX view 36
- CICSplex SM address space (CMAS)
 - administering 36
 - configuration 36
 - context for CICSplex SM commands 12
 - data repository 97
 - defining 36
 - definitions 35
 - links to other CMASs 25
 - maintenance point 28
 - NetView 11
 - performance considerations 24
 - preparing to set up 96
 - scope for CICSplex SM commands 12
 - use of ESSS 11
 - where to install 24
- CICSplex SM object model, recommended usage
 - business application services 54
 - monitoring 89
 - real-time analysis 81
 - workload management 71
- CICSplex SM, introduction to 1
- CICSSYS view 39
- CMAS 10
- CMAS view 36
- CMASD view 36
- CMASPLEX view 36
- CMTCMDEF view 36
- CMTMLND view 36
- CMTMLNK view 36
- CMTPLND view 36
- CMTPLNK view 36
- command checking, CICS 94
- configuration 101
 - CMAS 36
- configuration definitions
 - Starter Set 105
- CONNDEF view 39, 46
- context for CICSplex SM commands 12
- coordinating address space (CAS)
 - defining 35

- coordinating address space (CAS) *(continued)*
 - description of 12
 - installing 30
 - links to other CASs 31
 - preparing to set up 95
 - single point of control 31
 - Starter Set definitions 104
 - where to define 30
- copying definitions between CMASs 97
- CPLEXDEF view 36, 37
- CPLXCMAS view 36
- creating resource definitions 49

D

- data repository 5, 15, 97
- DB2CDEF view 46
- DB2EDEF view 46
- DB2TDEF view 46
- defining CAS-to-CAS links 35
- defining the CMAS configuration 35, 36
- defining the topology of a CICSplex 35, 37
- derived MONITOR data 84
- distributed model, dynamic routing 59
- distributed program link (DPL) 57
- DTRINGRP view 65
- dynamic installation of resources 52
- dynamic routing 4, 57, 69
 - distributed program links 57
 - models 58
 - program, EYU9XLOP 57
 - START commands 57
 - transactions 57
- dynamic transaction routing 57

E

- enterprise beans
 - workload balancing 62
 - workload separation 64
- entity types 92
- Environment Services System Services (ESSS) 11
- EVALDEF view 78
- external messages, RTA 74
- external notifications, RTA 74
- EXTRACT routine 15, 54
- EYU9XLOP 57

F

- FENODDEF view 46
- FEPOODEF view 46
- FEPRODEF view 46
- FETRDEF view 46
- FILEDEF view 46
- FSEGDEF view 46
- functions of CICSplex SM 1

G

- generic alert, SNA
 - CMAS considerations 24
- generic names 33, 93
- goal algorithm 62, 91
- group, CICS system 22, 38, 74

H

- hardware requirements 91
- hardware, prerequisite 92
- HTMLDEF view 46
- hub model, dynamic routing 58

I

- inconsistent set 49
- installation of resources
 - automatic 52
 - dynamic 52
- installation planning 91
- Installation Verification Procedures (IVPs) 104
- Interactive System Productivity Facility (ISPF) and CAS 12
 - using with a CAS 30
- intertransaction affinity 61
 - defining to CICSplex SM 64
 - identifying 69

J

- Job Control Language (JCL)
 - supplied with the Starter Set 103
- JRNLEDEF view 46
- JRNMDEF view 46

L

- links, communication
 - CAS-to-CAS 31
 - CMAS-to-CMAS 25
- local MAS 10
- LSRDEF view 46

M

- maintenance point CMAS 28
- managed application system (MAS)
 - definition 9
- MAP command 15
- map of CICS systems, producing 17
- MAPDEF view 46
- MAS agent code 10
- MAS resource monitoring (MRM)
 - automation and 77
 - description of 76
 - implementing 80
- MAS view 40
- models, dynamic routing 58
- MONDEF view 84

- MONGROUP view 84
- MONINGRP view 84
- MONINSPC view 85
- monitor definition 84
 - Starter Set 113
- monitor group 84
- monitor interval 84, 87
- monitor specification 85
- monitoring
 - CICS Monitoring Facility (CMF) 89
 - definition of 5
 - introduction to 83
 - monitor interval 84, 87
 - sample interval 84, 87
 - Starter Set samples 113
- MONSCOPE view 85
- MONSPEC view 85

N

- naming convention
 - CICSplex SM entities 33, 92
 - Starter Set 33, 92, 125
- NetView 11, 24
- notifications, external, RTA 74

O

- operating objects 14
- operations views 3

P

- PARTDEF view 46
- performance considerations
 - CMAS 24
 - CMAS-to-CMAS links 27
 - of CICSplex SM monitoring 86
- PERIODEF view 39, 80, 94
- PLEX view 35
- PLEXOVER view 35
- prerequisite hardware 92
- prerequisite software 91
- PROCDEF view 46
- PROFDEF view 46
- PROGDEF view 46

Q

- queue algorithm 62

R

- RACF (Resource Access Control Facility) 1
- RASGNDEF view 45
- RASINDSC view 45
- RASPROC view 45
- RDSCPROC view 46
- real-time analysis (RTA)
 - analysis point monitoring (APM) 77, 81

- real-time analysis (RTA) (*continued*)
 - AOC CICS Automation 74
 - definition of 4
 - external messages 74
 - implementing 80
 - introduction to 73
 - MAS resource monitoring (MRM) 76, 80
 - planning 79
 - SNA generic alert 74
 - Starter Set samples 116
 - system availability monitoring (SAM) 75, 80
- release synchronization ix
- remote MAS 10
- RESDESC view 46
- RESGROUP view 46
- RESINDSC view 46
- RESINGRP view 46
- resource assignment
 - migrating toward using 54
- resource checking 49, 94
- Resource Object Data Manager (RODM)
 - activating the interface to 89
- resource status facility
 - activating 89
- resource versions 42
- resources
 - automatic installation of 52
 - dynamic installation of 52
- retention period
 - monitoring 87
- RQMDEF view 46
- RTADEF view 78
- RTAGROUP view 78
- RTAINAPS view 78
- RTAINGRP view 78
- RTAINSPC view 78
- RTASCOPE view 78
- RTASPEC view 78

S

- sample interval
 - monitoring 84, 87
- samples, Starter Set 101
- scope for CICSplex SM commands 12
- SCREENS view 35
- security
 - BAS 52, 94
 - planning 93
 - RACF 1
- SESRES view 46
- SESSDEF view 39, 46
- simulating CICS security 94
- single point of control
 - definition of 2
 - role of CAS in providing 31
- single-system image (SSI)
 - definition of 2
 - role of CMAS in providing 10
- SNA generic alert 74
 - CMAS considerations 11, 24

- software requirements 91
- software, prerequisite 91
- staged implementation 34
- Starter Set
 - BAS samples 107
 - CAS definitions 104
 - configuration samples 105
 - definitions, supplied 104
 - installation requirements 102
 - JCL, supplied 103
 - monitoring samples 113
 - naming convention 125
 - purpose of 101
 - RTA samples 116
 - structure of 101
 - topology samples 105, 106
 - workload management samples 111
- starting CICSplex SM 96
- STATDEF view 78
- storage requirements 92
- SYSGRPC view 39
- SYSLINK view 40, 46
- system availability monitoring (SAM)
 - automation and 76
 - customizing external notifications 75
 - description of 75
 - implementing 80
- system group 21, 22

T

- TCPDEF view 46
- TDQDEF view 46
- TERMDEF view 46
- time zone 6, 94
- Tivoli Business Systems Manager
 - support for 15
- topology 2
 - administration 38
- topology definitions
 - Starter Set 105, 106
- TRANDEF view 46
- TRANGRP view 65
- TRNCLDEF view 46
- TSMDEF view 46
- TYPTMDEF view 46

U

- unauthorized access to CICSplex SM, preventing 94
- updating the configuration 34

V

- validating resources 49
- view
 - ACTDEF 78
 - ADMCONFIG 36
 - ADMTOPOL 38
 - APCMAS 78
 - APSPEC 78

view *(continued)*

- BATCHREP 97
- CASACT 35
- CASDEF 35
- CASDEFD 35
- CICSGRP 39
- CICSPLEX 36
- CICSSYS 39
- CMAS 36
- CMASD 36
- CMASPLEX 36
- CMTCMDEF 36
- CMTCMLND 36
- CMTCMLNK 36
- CMTPMLND 36
- CMTPMLNK 36
- CONNDEF 39, 46
- CPLEXDEF 36
- CPLXCMAS 36
- DB2CDEF 46
- DB2EDEF 46
- DB2TDEF 46
- DTRINGRP 65
- EVALDEF 78
- FENODDEF 46
- FEPOODEF 46
- FEPRODEF 46
- FETRGDEF 46
- FILEDEF 46
- FSEGDEF 46
- HTMLDEF 46
- JRNLDDEF 46
- JRNMDEF 46
- LSRDEF 46
- MAPDEF 46
- MAS 40
- MONDEF 84
- MONGROUP 84
- MONINGRP 84
- MONINSPC 85
- MONSCOPE 85
- MONSPEC 85
- PARTDEF 46
- PERIODEF 39, 80, 94
- PLEX 35
- PLEXOVER 35
- PROCDEF 46
- PROFDEF 46
- PROGDEF 46
- PRTNDEF 46
- RASGNDEF 45
- RASINDSC 45
- RASPROC 45
- RDSCPROC 46
- RESDESC 46
- RESGROUP 46
- RESINDSC 46
- RESINGRP 46
- RQMDEF 46
- RTADEF 78
- RTAGROUP 78

view (*continued*)

- RTAINAPS 78
- RTAINGRP 78
- RTAINSPC 78
- RTASCOPE 78
- RTASPEC 78
- SCREENS 35
- SESRES 46
- SESSDEF 39, 46
- STATDEF 78
- SYSGRPC 39
- SYSLINK 40, 46
- TCPDEF 46
- TDQDEF 46
- TERMDEF 46
- TRANDEF 46
- TRANGRP 65
- TRNCLDEF 46
- TSMDEF 46
- TYPTMDEF 46
- WLMDEF 66
- WLMGROUP 66
- WLMINGRP 66
- WLMINSPC 66
- WLMSCOPE 66
- WLMSPEC 66

W

- Web User Interface 7, 11
- WLMDEF view 66
- WLMGROUP view 66
- WLMINGRP view 66
- WLMINSPC view 66
- WLMSCOPE view 66
- WLMSPEC view 66
- workload definition 66
 - Starter Set 111
- workload functions of CICSplex SM 57
- workload group 66
- workload management (WLM)
 - definition of 4
 - intertransaction affinity 61, 64
 - introduction to 57
 - Starter Set samples 111
 - workload balancing 61, 69
 - abend compensation 62
 - goal algorithm 62
 - queue algorithm 62
 - workload separation 63, 70
- workload specification 66

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

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. A current list of IBM trademarks is available on the Web at Copyright and trademark information at www.ibm.com/legal/copytrade.shtml.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other product and service names might be trademarks of IBM or other companies.

Sending your comments to IBM

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM.

Feel free to comment on what you regard as specific errors or omissions, and on the accuracy, organization, subject matter, or completeness of this book.

Please limit your comments to the information in this book and the way in which the information is presented.

To ask questions, make comments about the functions of IBM products or systems, or to request additional publications, contact your IBM representative or 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, to this address:

IBM United Kingdom Limited
User Technologies Department (MP095)
Hursley Park
Winchester
Hampshire
SO21 2JN
United Kingdom

- By fax:
 - From outside the U.K., after your international access code use 44-1962-816151
 - From within the U.K., use 01962-816151
- Electronically, use the appropriate network ID:
 - IBMLink: HURSLEY(IDRCF)
 - Internet: idrcf@hursley.ibm.com

Whichever you use, ensure that you include:

- The publication title and order number
- The topic to which your comment applies
- Your name and address/telephone number/fax number/network ID.



Product Number: 5655-M15

SC34-6459-03



Spine information:



CICS TS for z/OS

CICSplex SM Concepts and Planning

Version 3
Release 1