

CICS Transaction Server for z/OS  
5.6

*CICSplex SM Concepts and Planning*

**IBM**

**Note**

Before using this information and the product it supports, read the information in [Product Legal Notices](#).

This edition applies to the IBM® CICS® Transaction Server for z/OS®, Version 5 Release 6 (product number 5655-Y305655-BTA ) and to all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright International Business Machines Corporation 1974, 2023.**

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

---

# Contents

<b>About this PDF</b> .....	<b>V</b>
<b>Chapter 1. CICSplex SM overview</b> .....	<b>1</b>
Features of CICSplex SM.....	1
The CICSplex SM environment.....	4
The CICSplex.....	6
The managed application system (MAS).....	7
CICSplex SM address space (CMAS).....	8
Environment Services System Services (ESSS).....	8
CICSplex SM objects.....	8
The data repository.....	9
<b>Chapter 2. CICS management client interface (CMCI)</b> .....	<b>11</b>
How it works: CMCI REST API.....	13
How it works: CMCI GraphQL API.....	13
How it works: CICS bundle deployment API.....	20
CMCI security features: How CMCI authenticates clients.....	23
<b>Chapter 3. Designing your CICSplex SM environment</b> .....	<b>25</b>
Designing your CICSplexes.....	26
Identifying your CICS systems and regions.....	26
How many CICSplexes are there in your enterprise?.....	28
Identifying system groups.....	31
Locating CMASs.....	32
Where to install CMASs.....	32
CMAS-to-CMAS links.....	33
Locating the maintenance point CMAS.....	37
Planning the location of WUI servers.....	37
Naming your CICSplex SM entities.....	39
A staged implementation.....	41
<b>Chapter 4. Planning CICSplex SM setup</b> .....	<b>43</b>
Security planning for CICSplex SM.....	43
Time zone planning for CICSplex SM.....	44
Reuse of CICSplex SM object definitions.....	45
<b>Chapter 5. Defining the CICSplex SM configuration and topology</b> .....	<b>47</b>
Defining the CMAS configuration.....	47
Defining CICSplex topology.....	48
Preparing CICSplex definitions.....	48
Planning CICS system definitions.....	48
Preparing CICS system group definitions.....	49
Where next?.....	51
<b>Chapter 6. Administering resources with CICSplex SM</b> .....	<b>53</b>
Managing resources using Business Application Services (BAS).....	53
What is Business Application Services (BAS)?.....	53
Managing BAS.....	58
Migrating your resource definitions.....	61

Defining resources.....	62
Using migration form BAS.....	63
Using full-function form BAS.....	63
Installing applications and resources.....	64
Security considerations.....	65
Planning for BAS.....	65
Implementing BAS.....	66
Where next?.....	67
<b>Chapter 7. Monitoring using real-time analysis (RTA).....</b>	<b>69</b>
What is CICSplex SM real-time analysis?.....	69
The RTA external notifications.....	70
System Availability Monitoring (SAM).....	70
MAS resource monitoring (MRM).....	72
Analysis point monitoring (APM).....	73
Managing RTA.....	73
Planning for RTA.....	76
Implementing RTA.....	76
Implementing system availability monitoring (SAM).....	76
Implementing MAS resource monitoring (MRM).....	77
Implementing analysis point monitoring (APM).....	77
A recommended approach.....	77
<b>Chapter 8. Collecting statistics using CICSplex SM monitoring.....</b>	<b>79</b>
What is CICSplex SM monitoring?.....	79
Managing monitoring functions.....	80
Planning CICSplex SM monitoring.....	81
Implementing CICSplex SM monitoring.....	83
Activating the resource status facility.....	83
Using the CICS Monitoring Facility (CMF) with CICSplex SM.....	84
A recommended approach.....	84
Where next?.....	84
<b>Notices.....</b>	<b>85</b>
<b>Index.....</b>	<b>91</b>

## About this PDF

---

This PDF provides a general introduction to CICSplex<sup>®</sup> SM. It is for system designers, planners, and programmers responsible for preparing an enterprise for the introduction of CICSplex SM.

It describes:

- At a high level, how CICSplex SM works, and introduces some new concepts
- The main functions and benefits of CICSplex SM as a CICS system management tool
- How to approach the design of a CICSplex SM configuration
- How to start defining your CICSplex SM environment
- How to plan for use of CICSplex SM Business Application Services, workload management, real-time analysis, and monitoring functions
- How to prepare for the installation of CICSplex SM.

These tasks are done in conjunction with the *Program Directory*.

For details of the terms and notation used, see [Conventions and terminology used in CICS documentation in IBM Documentation](#).

### **Date of this PDF**

This PDF was created on 2024-04-22 (Year-Month-Date).



---

# Chapter 1. CICSplex SM overview

The CICSplex System Manager element of CICS Transaction Server for z/OS is a system management tool that enables you to manage multiple CICS systems from a single control point.

Enterprises in which CICSplex SM might be needed range from those running 10 or 15 CICS regions, to those running two or three hundred (or more) CICS regions: in the latest z/OS sysplex environment, having such large numbers of CICS regions 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 regions are managed by a discrete controlling region that has no associated terminals or use application elements or components. There can be multiple instances of this managing region.
- User access to CICSplex SM functions is provided using 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 procedures and data
- CICSplex SM 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.

---

## Features of CICSplex SM

CICSplex SM incorporates a number of system management functions, including a real-time single-system image, a single point of control, business application management, and operations for the entire CICSplex.

### **• 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 the transaction processing environment of your enterprise. CICSplex SM creates an inventory of the CICS systems and resources and maps the relationships between them; this process is called the CICSplex SM *topology*. 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 can each 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)** enables 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. You can reuse a resource definition and associate it with more than one group as needed. BAS enables you to associate your resources according to their use in your enterprise. You can 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.

See [Managing resources using Business Application Services \(BAS\)](#).

- **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 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 use the Web User Interface *action buttons* to affect the status of the displayed resources. The responses are displayed in panels known as *operations views*, that can summarize related facts and provide access, using links, 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 [CICS operations views](#) and [The CICSplex SM Web User Interface](#) for more information about operations views.

- **Management of your workloads**

CICSplex SM 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)
  - Open Network Computing (ONC) RPCs
  - Any function that issues an EXEC CICS LINK PROGRAM request
- Transactions 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 [Overview of BTS](#).

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 workload routing functions of workload management
- User replaceable dynamic routing program EYU9XL0P
- Reduction in link definitions if you also use BAS
- Cooperation between BAS and BTS in the management of your business environment

You can write your own program to replace EYU9XL0P, and use it with the CICS-supplied user-replaceable programs DFHDYP and DFHDSRP, to handle your dynamic routing requirements.

See [Managing workloads through CICSplex SM](#) for more information about dynamic routing and workload management.

• **Automated exception reporting for CICS resources**

CICSplex SM 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 the status of a resource. With real time analysis you can take actions without having to use a separate automation product.

The RTA functions of CICSplex SM are described in greater detail in [Monitoring using real-time analysis \(RTA\)](#). For full details, see [Real-time analysis in Monitoring](#).

• **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 in a set of CICS systems.

The monitoring functions of CICSplex SM are described in more detail in [Collecting statistics using CICSplex SM monitoring](#). For full details, see [Real-time analysis in Monitoring](#).

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

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

- Access information about CICS and CICSplex SM resources.
- Invoke the services of CICSplex SM.

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

- Assembly
- PL/I
- COBOL
- C

In addition, a REXX runtime interface is available.

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 [CICSplex SM commands overview](#).

- **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 8](#).

- **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, the CICSplex SM API, or the batched repository-update facility.

- **The batched repository-update facility**

With the batched repository-update facility you can 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. For details, see [Administering CICSplex SM](#).

- **Management of time-dependent activity**

Much of CICSplex SM 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 might 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 [Administering CICSplex SM](#).

## The CICSplex SM environment

---

Before you begin to define your CICSplex SM environment, you must understand the components of CICSplex SM, either because you have to define them or because they can affect the design of your environment.

### What is a CICSplex?

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 SM is a management domain, made up of those CICS systems for which you want to establish a single system image (SSI).

[Read more...](#)

### What is a MAS?

Each CICS region that is managed by CICSplex SM is called a *managed application system* (MAS). You define and manage MASs as part of a CICSplex. Every MAS in a CICSplex is managed by a *CICSplex*

SM address space (CMAS). More than one CMAS can manage the MASs in a CICSplex, but a MAS can connect only to one CMAS at any given time.

[Read more...](#)

### What is a CMAS?

In every CICSplex, there is one *CICSplex SM address space* (CMAS) that is defined as the *maintenance point*. The maintenance point CMAS is responsible for maintaining the data integrity of the objects in every data repository by synchronizing its data repository with the data repository of other CMASs. It performs this synchronization by using CMAS-to-CMAS links, which are typically used for routing management commands and data between CMASs.

[Read more...](#)

### What is CMCI?

The CICS management client interface (CMCI) is a system management application programming interface for use by HTTP client applications including IBM CICS Explorer. You can use this interface to develop HTTP client applications that manage installed and definitional CICS and CICSplex SM resources in CICS regions in a CICSplex or CICS resources in stand-alone CICS regions.

[Read more...](#)

### What are the tools that you can use to administer CICSplex SM?

All CICSplex SM components, resources, system management requirements, and the relationships between them, are held as *objects* in a *data repository*. You can manage these objects using one or more of the provided interfaces:

- **CICS Explorer®**

CICS Explorer is an Eclipse-based tool that you can use to create, install, and manage the objects in the data repository. It requires a [CMCI or SM Data connection](#) to connect to CICS regions.

[Read more...](#)

- **The CICSplex SM Web User Interface (WUI)**

The CICSplex SM WUI provides a customizable, platform-independent web interface to create, install, and manage the objects in the data repository. With the WUI, you can access CICSplex SM through standard web browser software to perform operational and administrative tasks such as monitoring and controlling the resources of CICS systems, defining and maintaining the CICSplex SM configuration, specifying BAS, WLM, RTA, and monitoring requirements, and so on. The web browser client contacts a WUI server by an HTTP request via the CICS Web Interface.

The WUI server is a dedicated CICS region that runs as a CICSplex SM local MAS and communicates with the managed resources via the CMAS to which it is connected. You can have more than one WUI server active; for example, you might have a requirement for different languages to be used or different systems available to different servers.

**Note:** The Web User Interface server code must be at the same release level as the CICS region on which it runs and the CMAS to which it is connected.

All the menu and view definitions are stored on a server repository. There is one repository for each WUI server. The menu and view definitions can be exported for backup purposes, for distributing definitions to other servers, and for transferring menus and views when upgrading to a new product release.

[Read more...](#)

- **The batched repository-update facility**

The batched repository-update facility provides a batch job to create CICSplex SM resource definition objects.

The batched repository-update facility can help you in the migration of your CICS definitions to CICSplex SM. You can use the EXTRACT command of the CICS CSD update batch utility program

DFHCSDUP 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.

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.

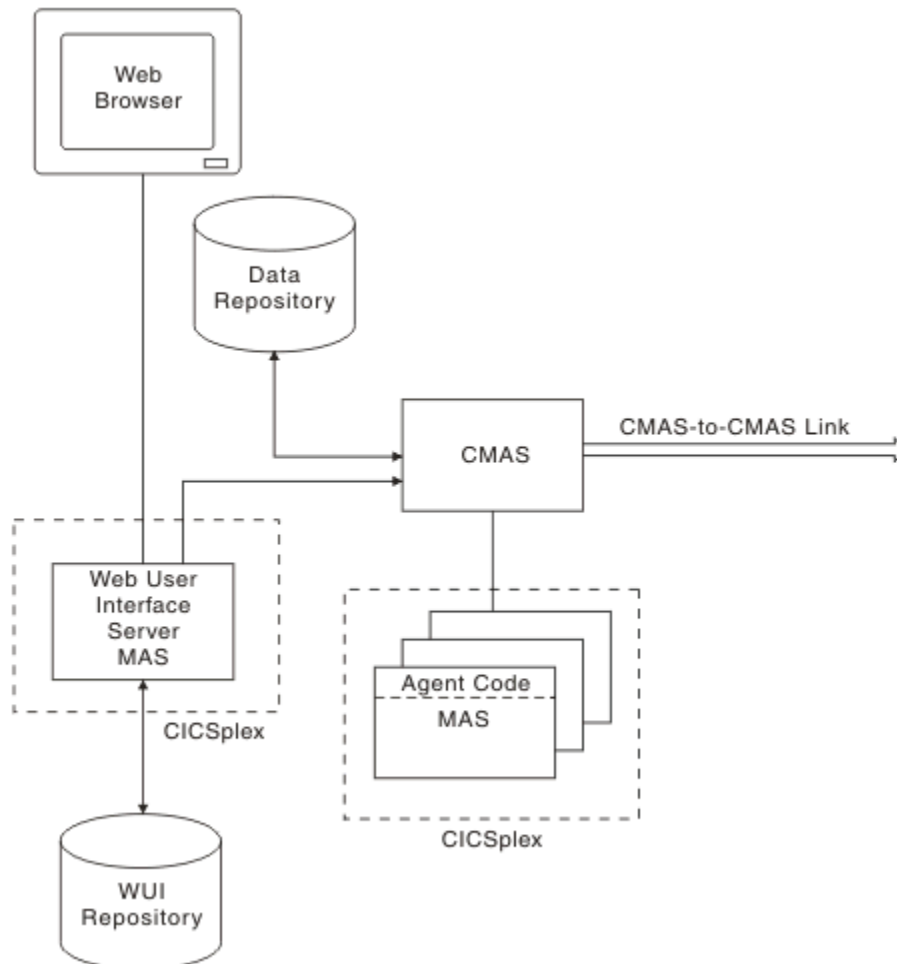


Figure 1. Key components of a CICSplex SM configuration

### Related information

[Setting up CICSplex SM](#)

## The CICSplex

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 CICS systems. 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 in a single system image manner.

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)*.

For details of the supported CICS releases that CICSplex SM can manage, see the information about upgrading CICSplex SM in [Upgrading](#).

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

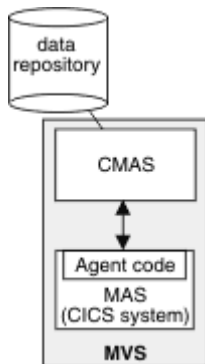


Figure 2. MASs and their CMAS

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](#).

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 (as shown in [Figure 1 on page 6](#)) and is communicated to the CMAS via the Environment Services System Services (ESSS) address space (which is described in [Environment Services System Services \(ESSS\)](#)).

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.

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 7](#)).

A CMAS is a CICS Transaction Server for z/OS 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 system address space that provides z/OS services to CICSplex SM components. In particular, ESSS owns all of the MVS/ESA data spaces on an z/OS 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 z/OS image as the CMAS.

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

## 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 CMAS data repository. [Figure 3 on page 9](#) shows how the CICSplex SM objects relate to each other.

This figure contains a high-resolution graphic that is not supported in this display format. To view the graphic, please use the CICS Information Center.

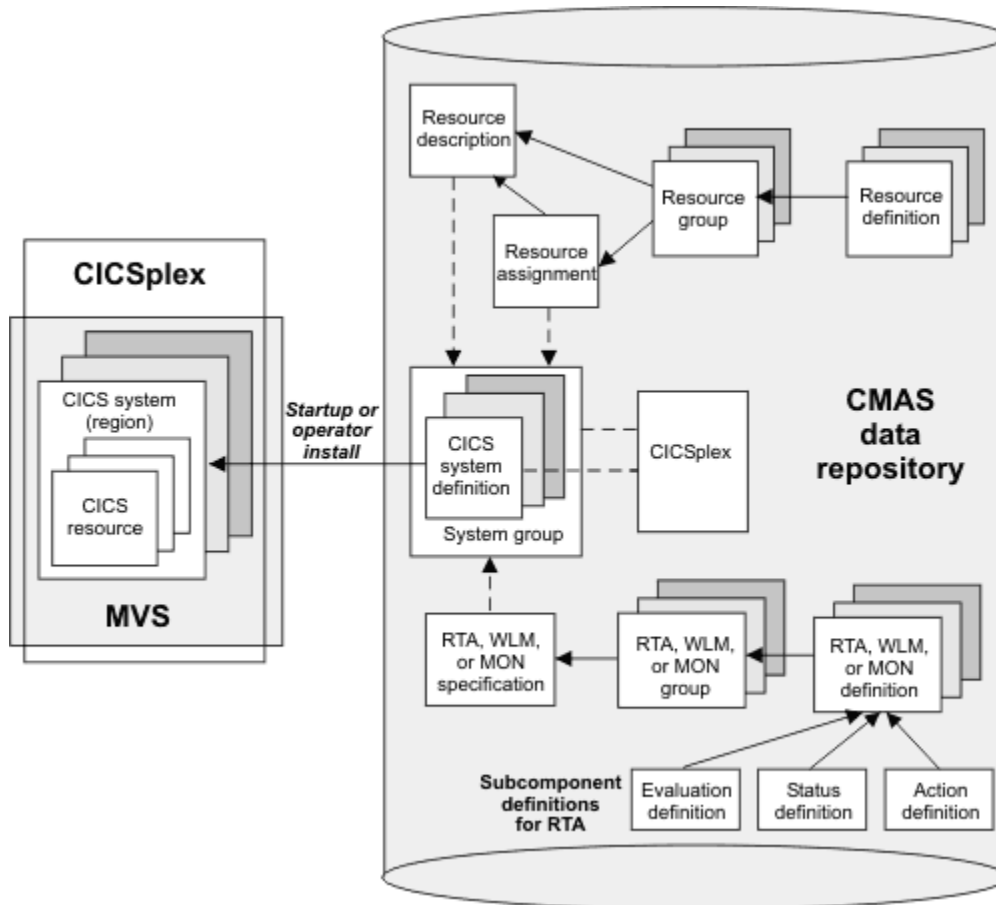


Figure 3. The CICSplex SM object model

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 [Designing your CICSplex SM environment](#); information on defining these objects is in [Defining the CICSplex SM configuration and topology](#).
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](#). Information on defining BAS objects is in [Using full-function form BAS](#).
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](#), [Workload management resources](#), and [Managing RTA](#).

## 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 in 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.

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

- Using CICS Explorer. See [Working with resources in the CICS Explorer product documentation](#).
- Using WUI views. See [The CICSplex SM Web User Interface](#).
- Using the EYU9XDBT CICSplex SM definition utility.
- Using the batched repository-update facility (BATCHREP); see [The batched repository-update facility](#).
- Using the API

You can generate 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, from a WUI detail or tabular view. CICSplex SM displays the starting point and all the definitions that either refer to that definition or are referred to by it. See [How to map CICSplex SM definitions](#).



## Chapter 2. CICS management client interface (CMCI)

The CICS management client interface (CMCI) is a system management application programming interface (API) for use by HTTP client applications such as IBM CICS Explorer. The CMCI provides the CMCI REST API and the CMCI GraphQL API for clients to manage installed and definitional system resources, and the CICS bundle deployment API to deploy bundles into single CICS development regions with Maven and Gradle plug-ins. It also supports advanced client authentication, including multifactor authentication (MFA).

Depending on your environment, you can set up the CMCI in either a CICSplex SM environment or in a single CICS region that is not managed by any CICSplex SM.

### CMCI in a CICSplex SM environment

Allows you to manage definitional resources, such as CSD and BAS resources, and operational resources in all of the CICS regions managed by CICSplex SM, with an HTTP client.

### CMCI in a single CICS region

Allows you to manage only the CSD and operational resources associated with that region through an HTTP client, and the context is specified as the application ID of that CICS region.

When a single CICS region is configured with the CMCI, it becomes a CICS System Management Single Server (SMSS).

### What is the CMCI JVM server?

The CMCI JVM server is a Liberty server. It is an optional, but highly recommended component of the CMCI that enhances support for CMCI requests. In addition to the basic CMCI REST API, the CMCI JVM server performs client authentication, including the support for multifactor authentication (MFA). It also provides powerful capabilities of system management and application development, through the CMCI GraphQL API and the CICS bundle deployment API.

Table 1 on page 11 compares functions that are available to the CMCI configured with the CMCI JVM server with those available to the basic CMCI (that is, without the CMCI JVM server).

Function	CMCI with the CMCI JVM server (default in CICSplex SM)	Basic CMCI
<b>Authentication support</b>		
User ID / password	✓	✓
Certificate	✓	✓
PassTicket	✓	
MFA	✓	
<b>API support</b>		
REST API	✓	✓
GraphQL API	✓	
CICS bundle deployment API	✓ (Additional configuration required)	

The CMCI JVM server is used by default by the CMCI in a CICSplex SM environment. For CICSplex SM, the enablement of the CMCI JVM server is controlled by feature toggle `com.ibm.cics.cmci.jvmserver`. For an SMSS, it's controlled by the **CPSMCONN** system initialization parameter.

## CMCI REST API versus CMCI GraphQL API: What is it? And what's the difference?

The CMCI REST API and the CMCI GraphQL API are both HTTP-based application programming interfaces that can be used to develop HTTP client applications that manage installed and definitional CICS and CICSplex SM resources on CICS regions being managed by CICSplex SM.

The CMCI REST API is designed based on Representational State Transfer (RESTful) principles, so you need to retrieve data from multiple endpoints with fixed data structures. That means each client of that API needs to be built with understanding of how to derive the relationships between resources. In comparison, these relationships are a fundamental part of the GraphQL API, so this API can expose only a single endpoint with more flexibility. This means that in a single query request, a client can query many types of CICS resources across CICSplexes, and specify exactly what data it needs with explicitly expressed relationships between the resources.

For example, the GraphQL query in [Figure 4 on page 12](#) retrieves data about the local transactions and associated programs, including use counts, in all regions in all connected CICSplexes. To achieve the same effect with the CMCI REST API, you might first access an endpoint that returns the list of local transactions available, and then an endpoint that returns all the programs. Then your client code must be written to post-process these results to match up the local transactions and programs. With GraphQL, relationships within the queried resources are also more explicitly shown through the CMCI GraphQL API than through the CMCI REST API.

```
{
  cicsplexes {
    cicsResources {
      loctrans {
        records {
          name
          to_program {
            name
            useCount
          }
        }
      }
    }
  }
}
```

*Figure 4. CMCI GraphQL API Query requesting programs associated with local transactions*

## CICS bundle deployment API: What is it?

Through the CICS bundle deployment API, the CMCI supports deploying CICS bundles into a single CICS region.

This REST API receives a CICS bundle as a zip file over HTTP. The bundle will be unzipped, installed into, and enabled in the appropriate CICS region automatically. If a CICS bundle with the same name already exists, it will be disabled and discarded before the new bundle is installed.

The CICS bundle deployment API can increase Java developers' productivity by enabling them to see their application changes reflected in a running CICS region within seconds. Developers can also use the CICS-provided Gradle or Maven plug-in (`cics-bundle-maven-plugin` or `com.ibm.cics.bundle`) that leverages the API, to integrate CICS bundle build and deployment into a toolchain.

The API also enables Java developers to deploy bundles whilst the system programmer retains control. A functional ID or another user ID with sufficient access deals with the bundle lifecycle and interacts with zFS on behalf of developers.

For more information about the API, see [“How it works: CICS bundle deployment API” on page 20](#).

To configure CICS for the API, see [Configuring the CMCI JVM server for the CICS bundle deployment API](#).

### Related information

[Setting up CMCI](#)

## How it works: CMCI REST API

---

The CICS management client interface (CMCI) provides a REST application programming interface (API) for system management clients such as IBM CICS Explorer. The CMCI REST API can also be used in an automated process, by leveraging the [Ansible IBM z/OS CICS collection](#).

The CMCI REST API is supported through HTTP. The client initiates an HTTP request to the CMCI. If the interface determines that the request is valid, it constructs a CICSplex SM API command or, in the case of a stand-alone CICS region, a CICS system command. After running the command, the CMCI creates an HTTP response. If the request is successful, this takes the form of an HTTP 200 (OK) response and an XML feed containing a result set, which it passes back to the client. If the request is not successful, the response consists of a non-OK HTTP response code with details of the failure.

The format for CMCI HTTP requests and responses is based on the HTTP/1.1 protocol. See [The HTTP protocol](#) for more information about this protocol.

### How to make CMCI HTTP requests

A CMCI request takes the form of an HTTP header followed by a URI (Universal Resource Identifier) and, where appropriate, an XML body containing details of any changes to be made to CICS or CICSplex SM resources.

The header incorporates one of the following HTTP methods:

#### **DELETE**

Removes resources from the CICSplex SM data repository, removes resources from the CICS system definition data set (CSD), or discards installed resources.

#### **GET**

Retrieves information about resources in the CICSplex SM data repository, retrieves information about resources on the CSD, or retrieves information about installed resources.

#### **POST**

Creates resources on the CICSplex SM data repository or resources in the CSD.

#### **PUT**

Updates existing resources in the CICSplex SM data repository, updates existing resources in the CSD, or sets attributes and performs actions on installed resources. Also performs actions on CICSplex SM and CSD resources.

The URI includes the name of a CICS or CICSplex SM resource, and specifies a series of parameters that refine the scope and nature of the query to identify one or more instances of the specified resource. In a GET request, the URI also specifies whether the API retains or discards a set of results. If the API retains the results, a new request can act on the retained results without having to repeat the retrieval operation. You can also use subsequent requests to page through the retained results selecting one or more records at a time.

POST and PUT requests include an XML body. In a PUT request the body contains either details of the changes to be made to resource attributes, or the action to be performed on the targeted resources. In a POST request, the body incorporates the attribute values you want to give to the new resource instance.

GET and DELETE requests do not require an XML body. If additional parameters are required for a DELETE request, those parameters must be included in the URI and can optionally be added to the XML body.

### Find out more

[CMCI REST API reference](#) gives your details on the DELETE, GET, POST, and PUT methods, CMCI resource names, CMCI XML body elements, diagnostic aids, and so on.

## How it works: CMCI GraphQL API

---

The CICS management client interface (CMCI) provides a GraphQL application programming interface (API) for system management clients such as IBM CICS Explorer. The CMCI GraphQL API is supported

through HTTP. With the GraphQL API, a client can query many types of CICS resources across CICSplexes or regions in a single request. In the single query request, the client can specify exactly what data it needs about multiple CICS resources, with inherent relationships between the CICS resources explicitly expressed. For more information about GraphQL, see [Introduction to GraphQL](#).

#### Notes:

- The aggregation function in CICS Explorer is also supported by the CMCI GraphQL API in CICS TS. For more information, see [Configuring for CICS Explorer](#).
- The CMCI GraphQL API is supported in a CICSplex SM environment as of CICS TS 5.5, and in a single CICS region (SMSS) environment as of CICS TS 5.6 with APAR PH35122.
- To set up the CMCI GraphQL API in CICS, you need to configure a CMCI JVM server within the WUI region of a CICSplex or a single CICS region. For instructions, see [Setting up CMCI](#).

## What is a GraphQL query?

A simple GraphQL query request looks like this:

```
{
  cicsplexes {
    name
  }
}
```

Figure 5. Simple query requesting CICSplex names

At the root of the query is the `cicsplexes` field, which finds all the CICSplexes that the WUI server is connected to. The `name` field nested in the `cicsplexes` field requests the name of each CICSplex.

Query responses are returned as JSON objects, with the requested data enclosed in the value of the `data` field. The structure of the response follows that in the query.

```
{
  "data": {
    "cicsplexes": [
      {
        "name": "CICSPLX01"
      },
      {
        "name": "CICSPLX02"
      }
    ]
  }
}
```

Figure 6. Response to simple query about CICSplex names

In an SMSS region, the structure of a GraphQL query is different and looks like this:

```
{
  smssRegion {
    name
  }
}
```

Figure 7. Simple query requesting SMSS region name

The structure of the response is similar to the CICSplex, and looks like this:

```
{
  "data": {
    "smssRegion": {
      "name": "IYCWENSS"
    }
  }
}
```

Figure 8. Response to simple query about SMSS region name

In CICS TS, the GraphQL API has support for:

- Basic system topology (CICSplexes, Regions, System Groups)
- Querying regions for the resources installed in them
- Querying BAS repositories and CSD repositories for definitions
- Aggregating and grouping resources
- Navigating links between resources

To retrieve more information, add more fields to the query, including nested ones. See [“Sample queries” on page 17](#).

## How to make GraphQL API requests

The GraphQL API endpoint is at:

```
https://host:port/graphql
```

where *host* and *port* are the host name and the port number of your CMCI JVM server.

The GraphQL API accepts GET and POST requests.

### For GET requests:

A Content-Type: `application/json` header must be sent. The query is supplied by the `query` query parameter. The operation is supplied by the optional `operationName` query parameter.

For example, the simple query in [Figure 5 on page 14](#) can be sent by using the following URL:

```
https://host:port/graphql?query={cicsplexes{name}}
```

Likewise, for the simple query in [Figure 7 on page 14](#), a GET request is similar, and can be sent by using the following URL:

```
https://host:port/graphql?query={smsRegion{name}}
```

### For POST requests:

A Content-Type: `application/json` header must be sent. The body of the request must be a JSON-encoded object.

```
{
  "query": "query_body",
  "operationName": "operation_name"
}
```

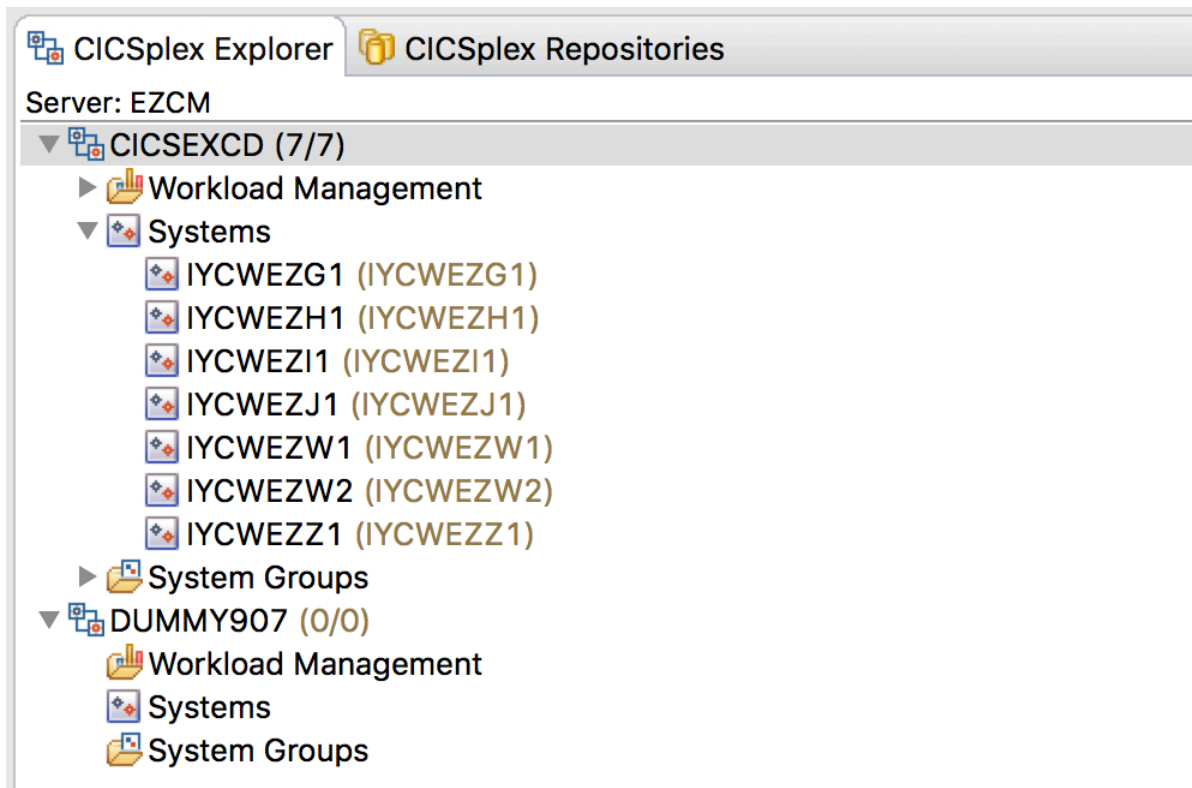
where only the `query` field is mandatory.

Alternatively, a Content-Type: `application/graphql` header can be sent on POST requests. In this case, the body of the request must be the GraphQL query itself, and no operation name can be specified.

See [“Sample queries” on page 17](#) for sample code of GraphQL queries.

## Example: the CICSplex Explorer view of CICS Explorer

The GraphQL API can provide a more responsive experience in user interfaces such as CICS Explorer. Take the **CICSplex Explorer** view of CICS Explorer as an example, consider what information is needed in CICS Explorer to build the **CICSplex Explorer** view:



At its most basic, you need to know what CICSplexes are in the environment, and which CICS regions are in those CICSplexes. With a REST API, you have to make multiple requests:

1. Ask for a list of CICSplexes in the environment.
2. For each CICSplex, ask for a list of the regions in that CICSplex.

Even this simple example demonstrates that the number of requests you have to make grows with the complexity of the information you ask for. When you consider adding region groups into the mix, and members of those region groups, even more information needs to be requested.

Here's how to ask for the information you need to populate the **CICSplex Explorer** view in the GraphQL API:

```
{
  cicsplexes {
    name
    regions {
      name
    }
  }
}
```

And here's the response:

```
{
  "data": {
    "cicsplexes": [
      {
        "name": "DUMMY907",
        "regions": []
      },
      {
        "name": "CICSEXCD",
        "regions": [
          {
            "name": "IYCWEZW2"
          },
          {
            "name": "IYCWEZG1"
          }
        ]
      }
    ]
  }
}
```

```

    "name": "IYCWEZW1"
  },
  "name": "IYCWEZH1"
  },
  "name": "IYCWEZI1"
  },
  "name": "IYCWEZJ1"
  },
  "name": "IYCWEZZ1"
}
]
}
]
}
}

```

The GraphQL supplies all the data you require in a single request with explicitly shown relationships between CICS resources. When enabled for the CMCI connection, it can drastically reduce the time you need to retrieve information through CICS Explorer. It also powers the aggregation function in CICS Explorer for Aqua 3.2 to provide aggregation and grouping of CICS resources.

## Sample queries

You can use GraphiQL, an online GraphQL visualization editor, to test your GraphQL queries or the samples. The URL to GraphiQL is:

```
https://host:port/graphiql
```

where *host* and *port* are the host name and the port number of your CMCI JVM server.

### GraphiQL tips:

- GraphiQL provides auto-completion and a built-in documentation explorer for GraphQL schema reference. You can display available field names by pressing Ctrl+Space.
- To easily differentiate queries in GraphiQL history, you can specify a unique query name by prefixing the query with `query QueryName`.

The following example queries the count of local files of all regions in all the connected CICSplexes, and the name of each CICSplex and region. It also has a query name `LocalFilesInRegionsInCICSplexes`.

```

query LocalFilesInRegionsInCICSplexes {
  cicsplexes {
    name
    regions {
      name
      cicsResources {
        locfile {
          count
        }
      }
    }
  }
}

```

You can add more attributes to be queried. This example queries all CICSplexes and all the regions in each CICSplex. Within each region, it retrieves the name, useCount, and status fields of all the local transactions.

```

{
  cicsplexes {
    name
    regions {
      name
      cicsResources {
        loctran {
          records {

```

```

      name
      useCount
      status
    }
  }
}

```

You can specify which CICSplex or CICS region to be queried. This example queries Region AORRGN in CICSplex PLEX1, retrieving the name, useCount, and status fields of all the local transactions in the region.

```

{
  cicsplex(name: "PLEX1") {
    name
    region(name: "AORRGN") {
      name
      cicsResources {
        loctran {
          records {
            name
            useCount
            status
          }
        }
      }
    }
  }
}

```

Removing the CICSplex and region specifications, this example queries all connected CICSplexes and the name, useCount, and status fields of all the local transactions in those CICSplexes.

```

{
  cicsplexes {
    name
    cicsResources {
      loctran {
        records {
          name
          useCount
          status
        }
      }
    }
  }
}

```

This example is similar to the previous one, except that it uses a filter to retrieve only transactions starting with CED.

```

{
  cicsplexes {
    name
    cicsResources {
      loctran(filter: {name: {value: "CED*"}}) {
        records {
          name
          useCount
          status
        }
      }
    }
  }
}

```

You can also query CICS definitions. This request queries the name and update attributes for all file definitions in the CICSplex data repository.

```

{
  cicsplex(name: "PLEX1") {
    drep {
      cicsDefinitions {

```



```

filedef {
  records {
    name
    update
  }
}
}
}
}
}
}
}
}
}
}

```

Similarly, this request queries the name for all pipeline definitions in the CSD for Region AORRGN in CICSplex PLEX1.

```

{
  cicsplex(name: "PLEX1") {
    region(name: "AORRGN") {
      csd {
        cicsDefinitions {
          pipedef {
            records {
              name
            }
          }
        }
      }
    }
  }
}
}
}
}
}
}
}
}
}

```

This query performs aggregation of all local files in each CICSplex, grouping them by common values of the name attribute and retrieving the count of aggregated records within each aggregation group, the name of each group, and the average, minimum, and maximum readCount within each group.

```

{
  cicsplexes {
    name
    cicsResources {
      locfile {
        groupBy(attribute: "name") {
          count
          aggregateRecord {
            name {
              value
            }
            readCount {
              average
              min
              max
            }
          }
        }
      }
    }
  }
}
}
}
}
}
}
}
}
}
}

```

The following example queries the current local transactions connected to the SMSS region. It retrieves records and displaying name, priority, status, tracing, purgeability, and deadlock timeout under each SMSS region name.

```

{
  smssRegion {
    name
    cicsResources {
      loctrans {
        records {
          name
          priority
          profile
          status
          tracing
          purgeability
          deadlockTimeout
        }
      }
    }
  }
}
}

```

```
}  
  }  
}
```

## How it works: CICS bundle deployment API

The CICS management client interface (CMCI) supports deploying CICS bundles into a single region through the CICS bundle deployment API. The API is supported by the CMCI JVM server that is configured either in the WUI region of the target region's CICSplex, or in the target region itself if it's a CICS System Management Single Server (SMSS).

### Table of contents

[“Overview” on page 20](#)  
[“How the API works” on page 21](#)  
[“Security model of the API” on page 22](#)  
[“How to make CICS bundle deployment API requests” on page 22](#)  
[“What's next” on page 23](#)

### Overview

The API is used at development time for Java developers to check their application changes in a running CICS region within seconds. The API also supports the CICS-provided Maven and Gradle plug-ins that can be used to integrate CICS bundle build and deployment into a developer's toolchain. For more information about CICS support for Maven and Gradle, see [Java support in CICS](#).

**Note:** CICS supports bundle installation from the CICS system definition data set (CSD) to a single region.

The CICS bundle deployment API enables Java developers to deploy bundles whilst the system programmer retains control. This is achieved by removing the need for developers to write bundles to zFS through FTP, or to install bundles from CSD. These actions are taken by the API, by using a functional ID or another user ID with sufficient access.

The diagram shows a typical scenario where multiple application developers push bundles to the API, which is configured in the WUI region of a CICSplex. For an SMSS environment, the API is configured in the CMCI JVM server of the target region.

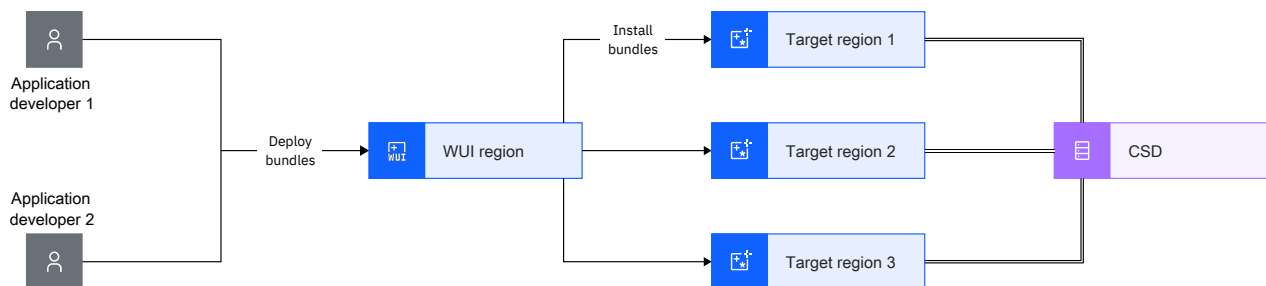


Figure 9. Multiple developers pushing bundles to API

This REST API receives a CICS bundle that contains the developer's application as a zip file over HTTP. The bundle will be unzipped, installed into, and enabled in the appropriate CICS region automatically. If a CICS bundle with the same name already exists, it will be disabled and discarded before the new bundle is installed.

Typically, the system programmer provides the Java developer with necessary parameters, for example, the CICSplex, CICS Region, CSD Group, and name of the BUNDLE definition. The system programmer then creates the BUNDLE definition, configured with the correct bundle directory attribute for the application that is being developed.

## How the API works

As described before, the API manages the bundle lifecycle and interacts with zFS on behalf of the developer after receiving the bundle. Usually the API is driven by a client such as the CICS-provided `cics-bundle-maven-plugin` or `com.ibm.cics.bundle`.

The API is configured with a **bundles directory**. This is a location on zFS that should be dedicated for the use of the API in managing bundles. Bundles pushed to the API are unzipped into the bundles directory and accessed by the CICS target region. This is transparent to developers and managed by the system programmer. Developers using this API do not need to interact with bundles on zFS.

For each application a developer is working on, the system programmer creates a **BUNDLE definition in the CSD**. The BUNDLE definition's `BUNDLEDIR` attribute must be configured with a path to the right version of the application uploaded to the bundles directory, for example, `/u/path/to/bundles/dir/ApplicationName_1.0.0`. The system programmer provides the BUNDLE definition name and CSD group for the developer to drive the API. Developers using this API do not need to install, disable, or discard the CICS BUNDLE resource.

The diagram shows a **bundle's lifecycle** from the time when it's published by an application developer until it gets installed into a CICS region. It's handled by the API automatically without users' intervention.

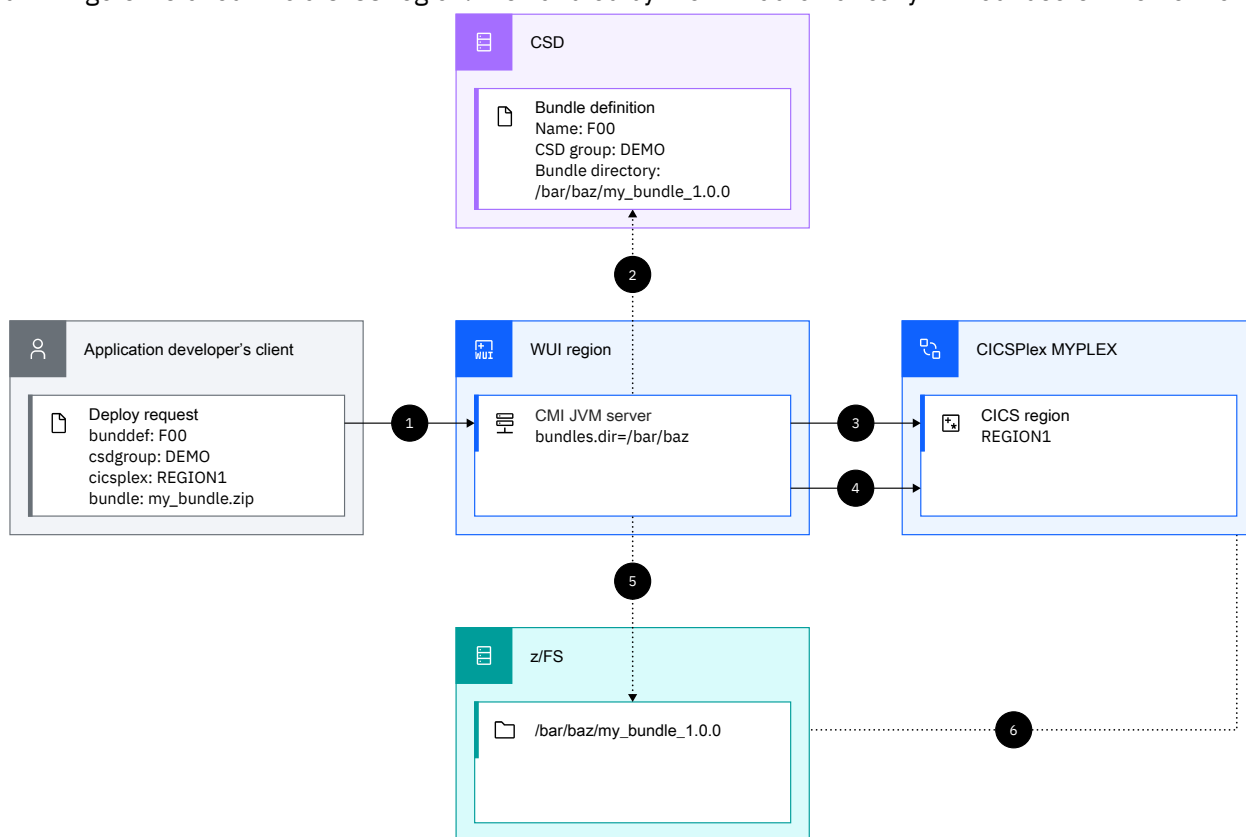


Figure 10. Bundle's lifecycle in the API

**1** The application developer publishes the application bundle through the CICS bundle deployment API.

### Validate

**2** The CMCI JVM server finds the BUNDLE definition in the target region's CSD and checks that the BUNDLE definition's bundle directory (`BUNDLEDIR`) attribute value is within the API's configured bundles directory.

### Uninstall

**3** The CMCI JVM server checks whether any previously installed bundle with the same name as the BUNDLE definition specified exists in the target region. Such a bundle is disabled and discarded as required.

4 The CMCI JVM server deletes any previous bundle with the same name and version from the bundles directory on zFS. Then, it unpacks the published bundle to the bundles directory.

### Install

5 The CMCI JVM server initiates a CSD install of the BUNDLE definition.

6 The target region reads the bundle from zFS and installs it.

## Security model of the API

A functional ID is introduced to ensure controlled access. As a result, application developers are able to deploy bundles through specific access to the CICS bundle deployment API, and without general access to manipulate bundles. Different users IDs are involved in the API's workflow as follows:

- The application developer is authenticated as normal, to make sure their user ID has access to the API.
- The task switches from the application developer's user ID to a configured deployment functional ID.
- The deployment functional ID writes the bundle to zFS and initiates a CSD install of the bundle to the target region.
- The region user ID of the target region reads the bundle from zFS during installation.

The security mechanism of the API requires additional configurations for the API to work:

- The system programmer must configure SAF correctly to enable bundle deployment using the API.
- The API works only in regions configured as `SEC=YES`.

For instructions, see [Configuring the CMCI JVM server for the CICS bundle deployment API](#).

## How to make CICS bundle deployment API requests

CICS provides a Maven and a Gradle plug-in that uses this API to publish bundles to CICS. Java developers can use it for bundle deployment at development time. For instructions, see [cics-bundle-maven](#). Alternatively, if you want to write your own tooling, the API can be invoked by using any standard REST client.

The format of API requests is based on HTTP/1.1. The API uses the POST request only, and accepts multipart form data. A `Content-Type: multipart/form-data` header must be sent. String parameters and the zipped bundle file are accepted as form data parts in the payload.

The API returns an HTTP response code and a JSON description of the outcome of the request. A response code of 4xx indicates that a user has incorrectly addressed the API. A response code of 5xx indicates that the system is set up incorrectly.

The API endpoint is served alongside your standard CMCI interface. For example, if your CMCI interface is available at `mycicshost.com` on port 4444, your API endpoint URL would be `https://mycicshost.com:4444/managedcicsbundles`.

Call the API with these parameters:

### **cicsplex**

Required only for a CICSplex SM environment. The name of the CICSplex that the target region belongs to.

### **region**

Required only for a CICSplex SM environment. The name of the region that the bundle should be installed into.

### **csdgroup**

The name of the CSD group where the source BUNDLE definition exists.

### **bunddef**

The name of the BUNDLE definition.

### **bundle**

The zip file that contains the bundle contents, as `content-type application/zip`. The `META-INF` directory should be in the root of the zip file.

For example, to call the API from your local workstation using the command-line tool cURL, your command line might look like this:

```
curl -i -X POST -u MYUSER -F cicsplex=MYPLEX -F region=MYREGION -F csdgroup=CSDGRP  
-F bunddef=MYBUND -F bundle=@c:/path/to/bundle_0.0.1.zip https://mycicshost.com:4444/  
managedcicsbundles
```

The CICS bundle deployment API unzips the content of the uploaded zip file without performing any codepage conversion. Each file in the bundle must be encoded in the correct codepage prior to being zipped.

## What's next

Configure your CMCI JVM server to enable the API. See [Configuring the CMCI JVM server for the CICS bundle deployment API](#)

## Related concepts

[Managing Java dependencies using Maven or Gradle](#)

## CMCI security features: How CMCI authenticates clients

When an HTTP system management client such as CICS Explorer attempts to sign on, the CMCI verifies the user credentials. The user credentials can be a user ID and password, a Passticket, an MFA token or a certificate. If the CMCI JVM server is enabled, it handles the authentication process. Authentication through a Passticket or an MFA token is only available with the CMCI JVM server.

### How the CMCI JVM server authenticates clients

[Figure 11 on page 23](#) illustrates the client authentication workflow based on CICS Explorer.

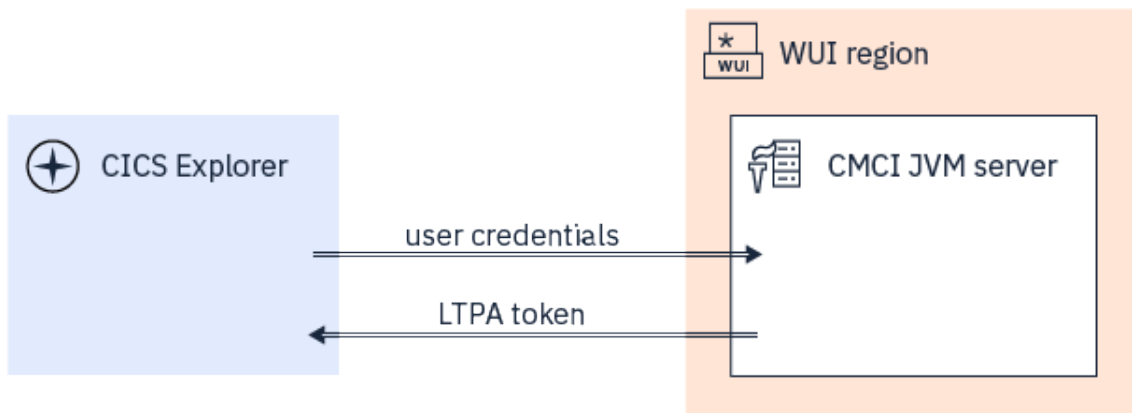


Figure 11. CMCI HTTP client authentication workflow

1. When a user logs on from CICS Explorer, CICS Explorer passes the user credentials to the CMCI JVM server. The user credentials can be a user ID and password, a Passticket, an MFA token or a certificate.
2. The CMCI JVM server validates the user credentials by using SAF interfaces to the external security manager (ESM) and generates an LTPA token.
3. The CMCI JVM server replies to CICS Explorer with the response and the LTPA token.

In subsequent requests, CICS Explorer will use the LTPA token to authenticate the user.

### Note:

- The LTPA token is a cookie; therefore, the HTTP client must accept cookies.

- Although a JVM server is used for the transport and authentication of the CMCI, most of the processing still occurs in the CICS core; therefore, do not expect increased specialty engine offload from the CMCI JVM server.

### **LTPA timeout**

An LTPA token has a fixed lifetime. It cannot be extended or renewed, even if a user is active in a session. Upon timeout, the user is logged out and must provide login credentials again to get a new token. The expiration time of the LTPA token is configurable. For instructions, see [Configuring LTPA in Liberty](#).

### **Sharing LTPA tokens**

With the single sign-on (SSO) configuration support in Liberty, you can set up Liberty to allow the sharing of LTPA tokens among multiple regions. HTTP client users can authenticate once and have access to other regions that share the same LTPA keys. For more information, see [Customizing SSO configuration using LTPA cookies in Liberty](#).

### **How CMCI without the CMCI JVM server authenticates clients**

If the CMCI JVM server is not used with the CMCI, the user is authenticated using a certificate or a basic authenticator in the HTTP header.

One-time-use tokens (such as MFA tokens and PassTickets) are not supported.

### **Find out more**

[Authentication overview](#) gives you an overview of the authentication process in Liberty and describes LTPA and SSO in details.

[Setting up CMCI](#) gives you configuration instructions.

---

## Chapter 3. Designing your CICSplex SM environment

Map out your enterprise to design a CICSplex SM environment that meets your system management requirements. Your design must include topology information, such as identifying the CICSplexes, CMASs, and WUI server regions that are required, as well as making other design decisions such as what naming convention to follow.

### Before you begin

To design a CICSplex SM environment, you must be familiar with the concepts and components of CICSplex SM

Be aware of some considerations that can help to avoid issues with the set-up or maintenance of CICSplex SM:

- When you apply service to CICSplex SM, PTFs that are applied to the ESSS are not intended to be downward-compatible with earlier maintenance levels at the same release. This means that all CMASs, MASs, WUI Server regions and API programs must run at the same maintenance level as the ESSS for their release. Otherwise, abends, data corruption, and unexpected results might occur. When you apply PTFs to CICSplex SM, it is essential that all ++HOLD ACTION items associated with the SMP/E maintenance are followed carefully.
- Additionally, consider the following guidelines when you design a CICSplex SM topology:
  1. Avoid running Production and non-Production (for example, test, development, or QA) regions on the same LPAR. All regions that run the same CICSplex SM release use one copy of the ESSS: that is, they share the ESS. As a result, applying a PTF to this shared ESSS requires an outage of both the Production and the non-Production regions.
  2. Connect the WUI server directly to the maintenance point CMAS (MP CMAS), so that they are both at the at the highest CICSplex SM release level. This configuration ensures that the WUI server uses the latest resource tables and WUI views, and simplifies upgrading procedures.

When you run a mixed release environment, you might need more than one WUI server. In that case, this advice still applies. Have one WUI server at the latest release level and connected directly into the latest release CMAS, and any other earlier release WUI servers connected into the appropriate release CMAS, if required.
  3. Connect the MP CMAS only to other CMASs and a WUI server. If MASs are connected directly into the MP CMAS, this configuration prevents the MP CMAS from being easily moved to a different LPAR if an outage occurs.
  4. Use a single MP CMAS to prevent complexity during upgrades. For example, this helps avoid a potential conflict of CICSplex SM release levels, which can cause CMASs to be isolated from the management of a CICSplex during an upgrade.

### Procedure

1. Decide how to group your CICS regions into system groups to identify what CICSplexes are required.

The CICSplex is the largest unit that you can manage in your CICSplex SM configuration.
2. Decide how many CMASs are required for each CICSplex and how to link them together.

The CMAS is the component of the CICSplex SM topology that is responsible for most of the work involved in managing and reporting on CICS regions and their resources. Each CICSplex must have at least one CMAS.
3. Decide how many WUI servers are required and how to link them to the correct CMAS.
4. Decide on a naming convention for the CICSplex SM components.

The naming convention must be meaningful and extendable.
5. Decide on an implementation strategy.

## Results

Following these steps results in a detailed topology map of your enterprise.

## What to do next

After you complete your map of the enterprise, you are ready to plan the installation of CICSplex SM to create your proposed configuration. Make sure that you keep your map up-to-date as you add more system groups or regions or make changes to the topology, because an accurate map makes it easier to maintain the CICSplex SM configuration and topology data.

## 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 first action when planning to install CICSplex SM is to identify the CICS systems or regions in your enterprise. You might 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.

The goal 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 you do not need to record details such as where specific processors are located. When you record the map, for example by using a graphics tool, ensure that you leave plenty of space so that you can update the map with CICSplex SM components as you work through this exercise.

In your initial map of the enterprise CICS systems, include every operating environment in which CICS is installed. Also show the following information:

- The current version and release of the operating systems in use
- 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 cannot fit all this detail on your map, record it separately from the graphical representation of the CICS systems.

[Figure 12 on page 27](#) shows an example of the type of map you should be aiming to produce.



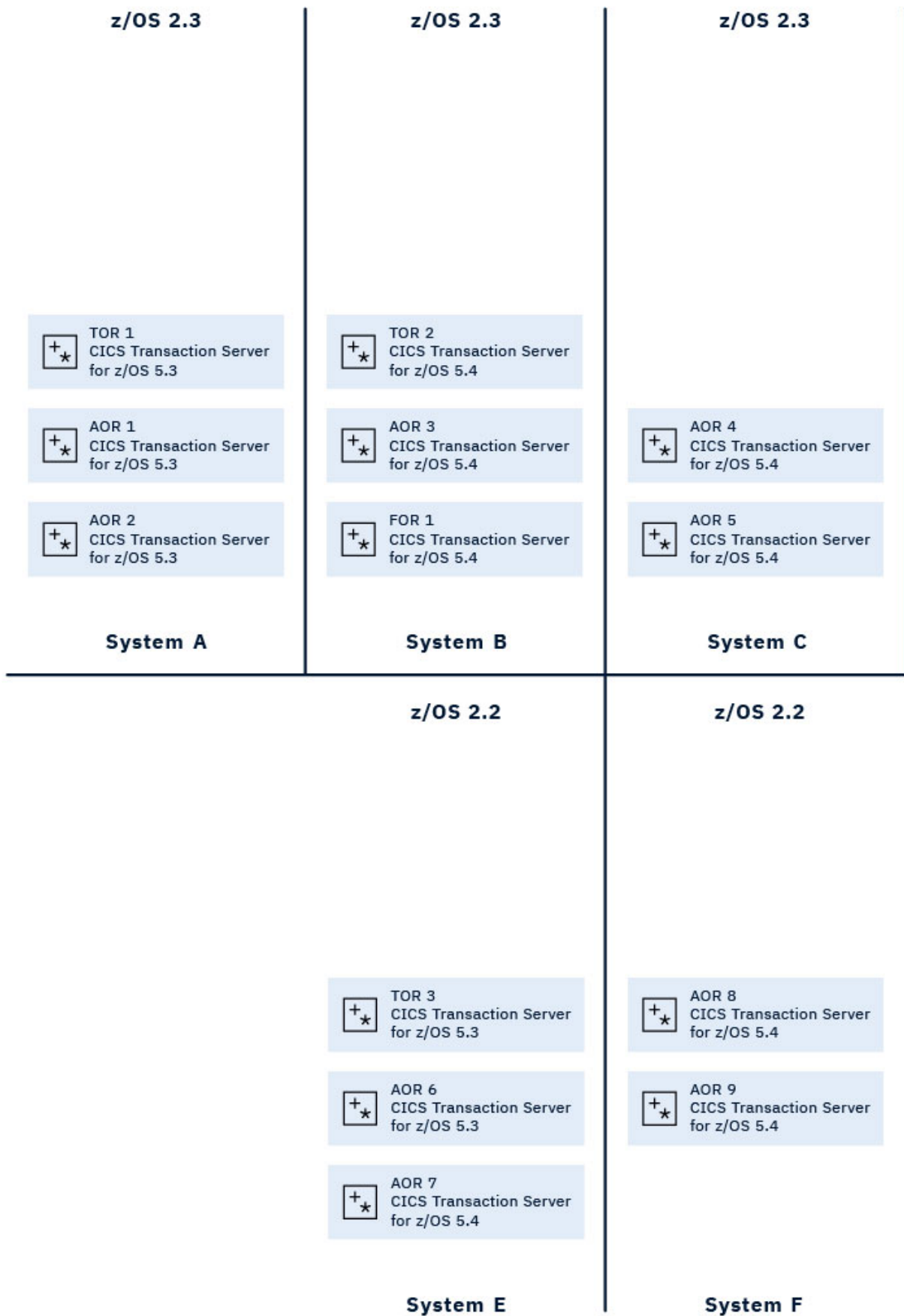


Figure 12. A map of the enterprise CICS systems

CICSplex SM can manage all supported CICS releases. The CICS systems become your managed application systems (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 should belong to each CICSplex.

You can have any number of CICSplexes. For example, you could define one CICSplex for:

- The entire enterprise
- Each geographical location
- Each business unit in the enterprise
- Each existing TOR-AOR-FOR configuration
- Each processor

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 in 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, keep a business application in 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.

### Step 1: Have a single CICSplex for the entire enterprise

The majority of enterprises find that having a single CICSplex best suits their system management goals.

The CICSplex is the largest single entity that CICSplex SM can manage. None of the CICSplex SM definitions or specifications can cross a CICSplex boundary. Furthermore, CICSplexes are mutually exclusive. No CICS system can belong to more than one CICSplex. Therefore, having a single CICSplex for the enterprise brings several advantages. For example:

- Providing the greatest opportunity for sharing and reusing BAS, WLM, RTA, and monitoring specifications and definitions.
- Giving you the most flexibility in managing the CICS workload when using the CICSplex SM workload routing functions.

- Giving a single system image (SSI) of the entire enterprise. 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 the best approach in every enterprise, either because implementation is not possible, or because it does not 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 is 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 in 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, consider defining multiple CICSplexes instead of one.

If you have decided that you need to define more than one CICSplex, it is probably obvious to you already which CICS system or CICS systems should belong to each. If it is not, consider revisiting your decision to have multiple CICSplexes because it suggests that you are trying to erect artificial barriers. As a final check, 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 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:

- Does your CICSplex organization mirror your enterprise organization? If your enterprise is structured as multiple, independent units, having multiple CICSplexes is probably the better approach. If it is 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?
- 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.

Remember that you can alter your decision. Ideally, you will 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.

In [Figure 13 on page 30](#), 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 divided between two CICSplexes.

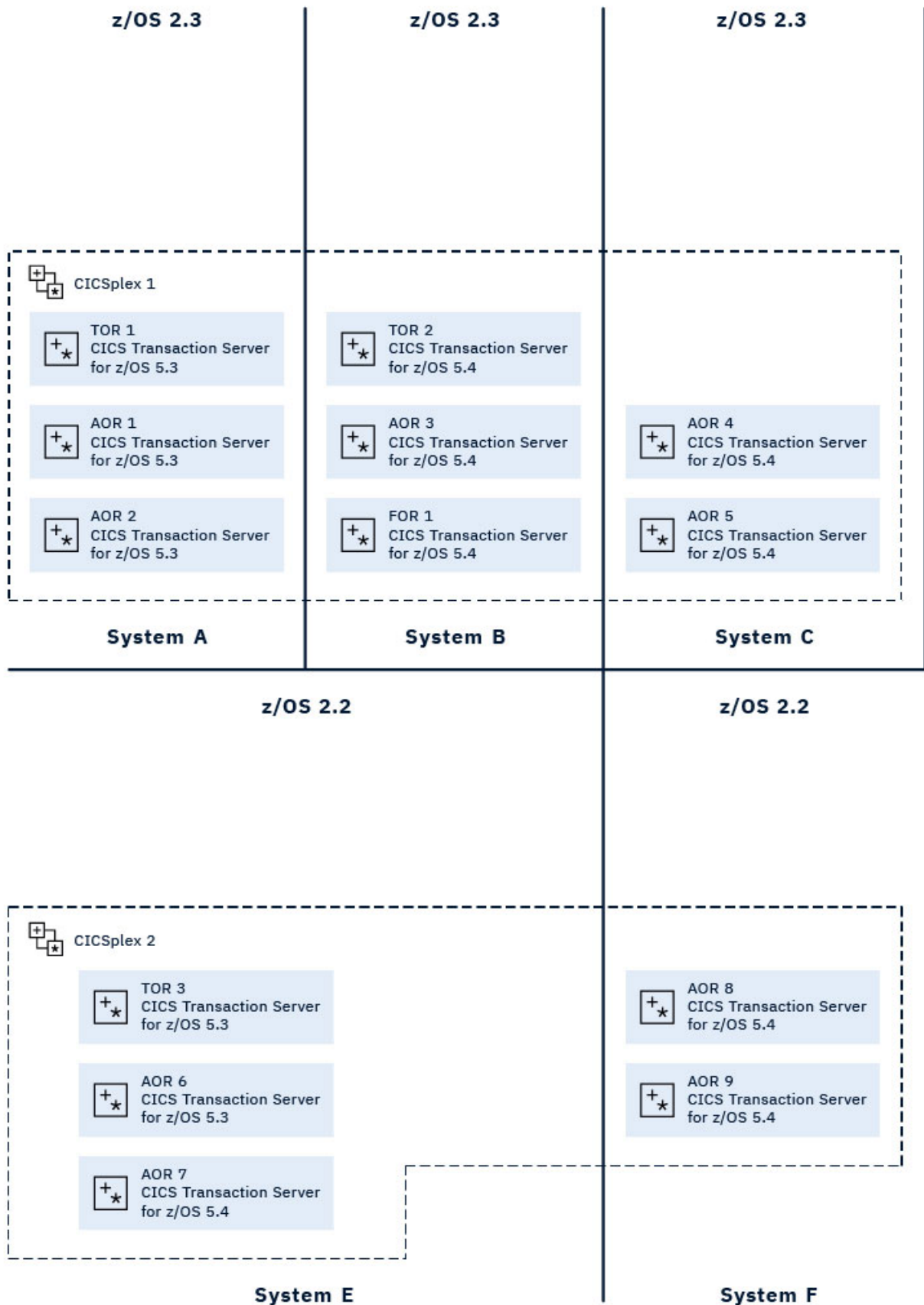


Figure 13. Identifying the CICSplexes in the example enterprise

## Identifying system groups

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.

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 in a CICSplex. However, because the CICS system group is a subset of the CICSplex, a system group cannot cross CICSplex boundaries.

In the example configuration in [Figure 13 on page 30](#), some suggested CICS system groups are as follows:

- 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 F
  - 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.

### 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

After you decide which CICS systems will be managed by CICSplex SM, and how they will 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 as follows:

- 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 SDFH\* libraries and the SEYU\* libraries in the CMAS must be the same level, and the CMAS must be appropriate to the level of CICSplex SM that you are running. See, [Upgrading CICSplex SM](#)
- 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.
- Install 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 these rules and recommendations are applied 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 14 on page 33 shows the example map updated to show the application of these CMAS rules and recommendations. A CMAS is to be installed on each MVS image. CMAS 4 can issue SNA generic alerts to the NetView instance on the same MVS image. CMAS 4 is the maintenance point CMAS (MP CMAS) for CICSplex 1, and CMAS 7 is the MP CMAS for CICSplex 2. That is, CMAS 4 will be the context CMAS when you define CICSplex 1, and CMAS 7 will be the context CMAS when you define CICSplex 2 to CICSplex SM.)

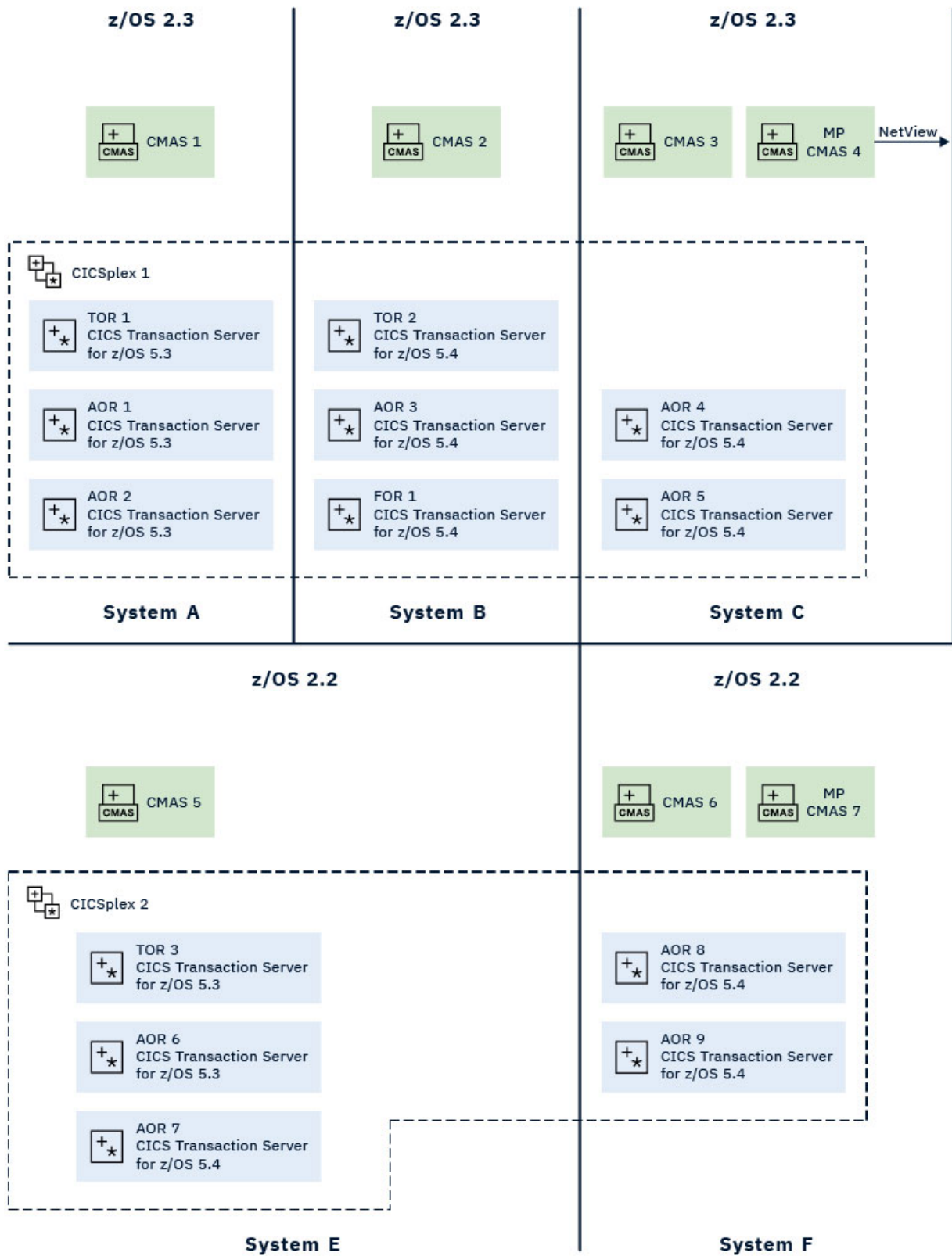


Figure 14. Adding CMASs to the example enterprise map

## 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 (by using CICS intercommunication methods) so



that data relating to the CICS systems belonging to the CICSplex is accessible by all relevant CMASs, and a single system image (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.

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.
- 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.

Figure 15 on page 34 illustrates the concept of direct and indirect CMAS-to-CMAS links. This CICSplex comprises six CICS systems; three systems are managed by CMAS 1, and three systems are managed by CMAS 5. Without a direct link between the two CMASs, CICSplex SM can navigate dynamically around its "network" to gather information relevant to the CICSplex. It could go through CMAS 4 or, if CMAS 4 is unavailable, it could go through CMAS 2 and CMAS 3. However, for optimum performance, a direct link should be added between CMAS 1 and CMAS 5.

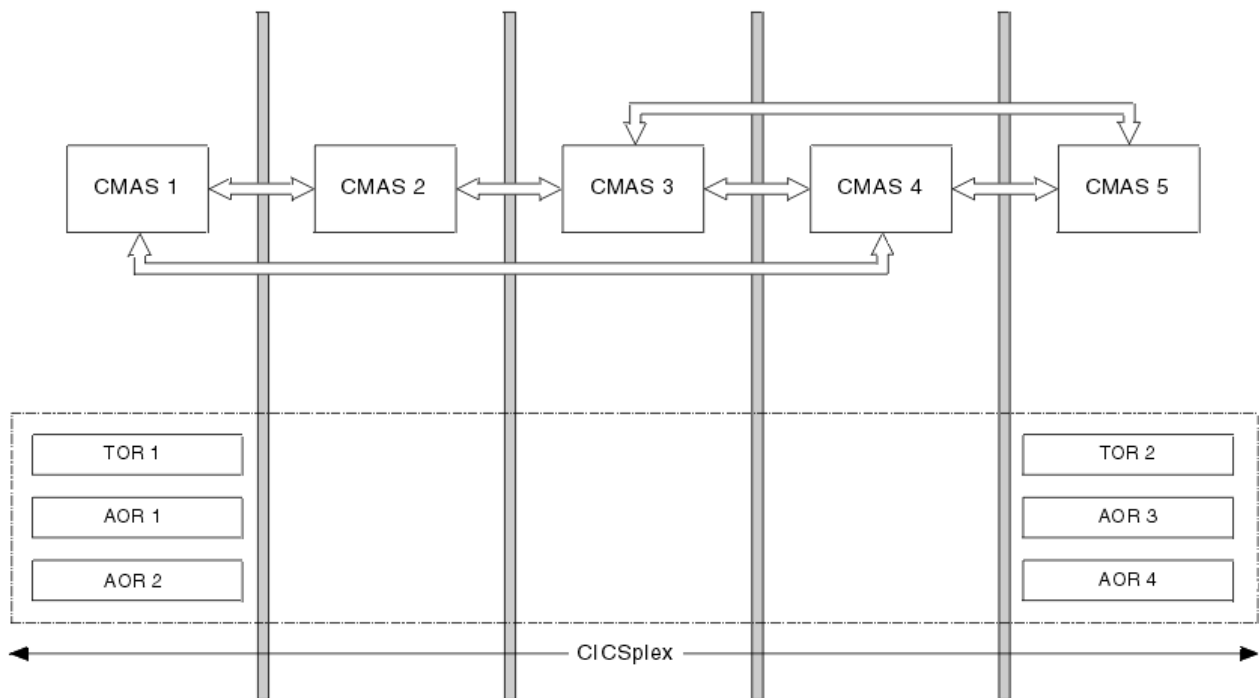


Figure 15. CMAS-to-CMAS links in a CICSplex

In the example enterprise configuration in Figure 16 on page 36, full CMAS-to-CMAS connectivity is established for each of the two CICSplexes. That is, for management of CICSplex 1, CMASs 1, 2, 3 and 4 are directly connected to each other; for management of CICSplex 2, CMASs 5, 6, and 7 are directly connected to each other. Thus, the following CMAS-to-CMAS links must be defined:



**CICSplex 1**

CMAS1 to CMAS2  
CMAS1 to CMAS3  
CMAS1 to CMAS4  
CMAS2 to CMAS1  
CMAS2 to CMAS3  
CMAS2 to CMAS4  
CMAS3 to CMAS1  
CMAS3 to CMAS2  
CMAS3 to CMAS4  
CMAS4 to CMAS1  
CMAS4 to CMAS2  
CMAS4 to CMAS3

**CICSplex 2**

CMAS5 to CMAS6  
CMAS5 to CMAS7  
CMAS6 to CMAS5  
CMAS6 to CMAS7  
CMAS7 to CMAS5  
CMAS7 to CMAS6

If you decided to not link CMAS 1 to CMAS 3 directly, this arrangement would meet the minimum CMAS-to-CMAS link requirement (by having an open chain of CMASs) and you would need to define two less links. Some information would be obtained indirectly. For example, information requested from System C about CICS resources on System A would be obtained indirectly through the adjacent CMAS, CMAS 2. 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 you must balance this against the overhead of setting up and maintaining the CMAS-to-CMAS links. In fact, if the example enterprise configuration had one CICSplex rather than two, full connectivity among the CMASs would be feasible. That is, you would need 42 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 have ten CMASs managing a single CICSplex, the number of links required is 90; for 15 CMASs, 210 links. So you must 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 16 on page 36 shows the CMAS-to-CMAS links in the example enterprise configuration. 18 links are defined, giving full connectivity among CMASs managing the same CICSplexes.

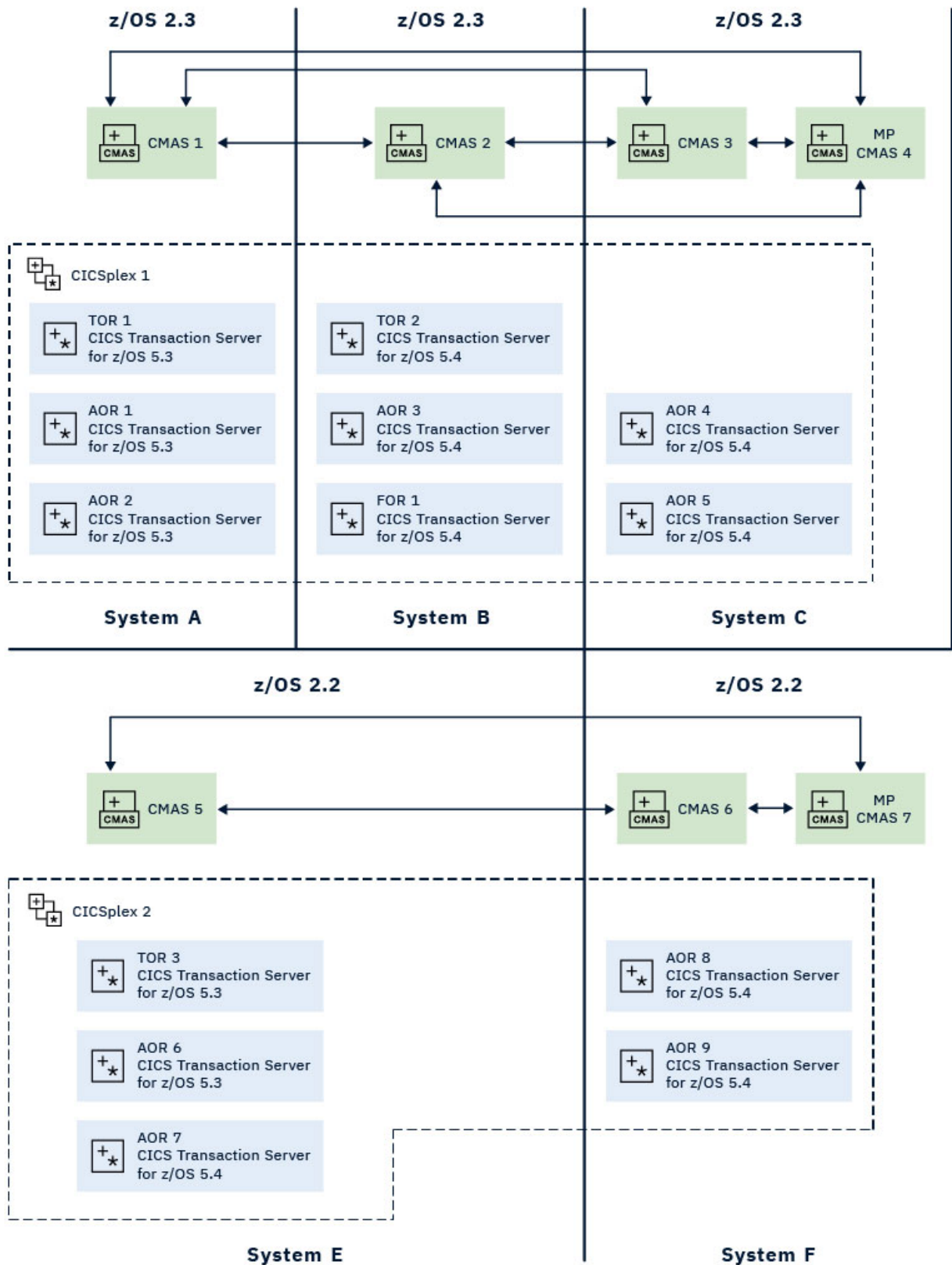


Figure 16. Adding CMAS-to-CMAS links to the example enterprise map

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.

## CMAS-to-CMAS links for multiple CICSplexes

The example enterprise configuration has two CICSplexes. To enable the CICSplex SM operator to 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, thus forming a chain of all of the enterprise CMASs.

## Locating the maintenance point CMAS

When a single CICSplex is managed by multiple CMASs, one of them is designated the *maintenance point* CMAS (MP CMAS).

The MP 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 MP CMAS is the CMAS that is the *context* value when the CICSplex is defined to CICSplex SM. Each CICSplex can have only one MP CMAS, but a single CMAS can be the MP CMAS for more than one CICSplex. If the MP CMAS is unavailable at any time, no changes to any CICSplex SM definitions can be made.

An MP CMAS can also function as a non-maintenance point CMAS for other CICSplexes.

**Note:** The CICSplex SM single point of control means that changes to the MP CMAS can be made from any system on which the CICSplex SM WUI is available.

Ideally, connect the MP CMAS only to other CMASs and a Web User Interface (WUI) server. Do not connect the MP CMAS to other CICS regions. This configuration means that the MP CMAS can easily be moved between MVS images, or have maintenance applied, without affecting other regions.

## Planning the location of WUI servers

---

The location and number of WUI server regions in your enterprise depend on your availability requirements and globalization support.

### Before you begin

You must have planned where to install the CMASs and how to link them together before planning where to install the WUI server regions.

### About this task

The WUI server is a CICS region that acts as a CICSplex SM application, using the API to view and manage objects in the data repositories of CMASs.

### Procedure

1. Decide if you want a WUI server to be available in every MVS image.

If you have a WUI server on each MVS image, you do not have to connect the WUI servers directly to the CMAS on each MVS image.

**Tip:** If you define a separate CICSplex for the WUI servers, the number of CMAS maintenance points across your enterprise are kept to a minimum. It also separates the WUI servers from statistics generated from CICSplexes that are running CICS applications.

2. Decide what national languages the WUI must support.

If you want to display the WUI in more than one language, you require a WUI server for each language.

3. Decide what connections are required from each WUI server region to the CMASs.
  - The CMAS to which the WUI server connects must be managing all the CICSplexes to which the WUI server requires access. However, it is not necessary for the CMAS to which the WUI connects to be managing any of the MASs in these CICSplexes.
  - The CMAS and the WUI server that you want to connect must be at the same release level of CICS Transaction Server.
4. Update your topology map to include the WUI server regions.

**Example**

On system F, a WUI server is connected to the maintenance point, CMAS 7, and is in a separate CICSplex called CICSplex 4. The WUI server and CMAS 7 are both at the same level of CICS Transaction Server. CMAS 7 is the maintenance point for both CICSplex 2 and CICSplex 4.

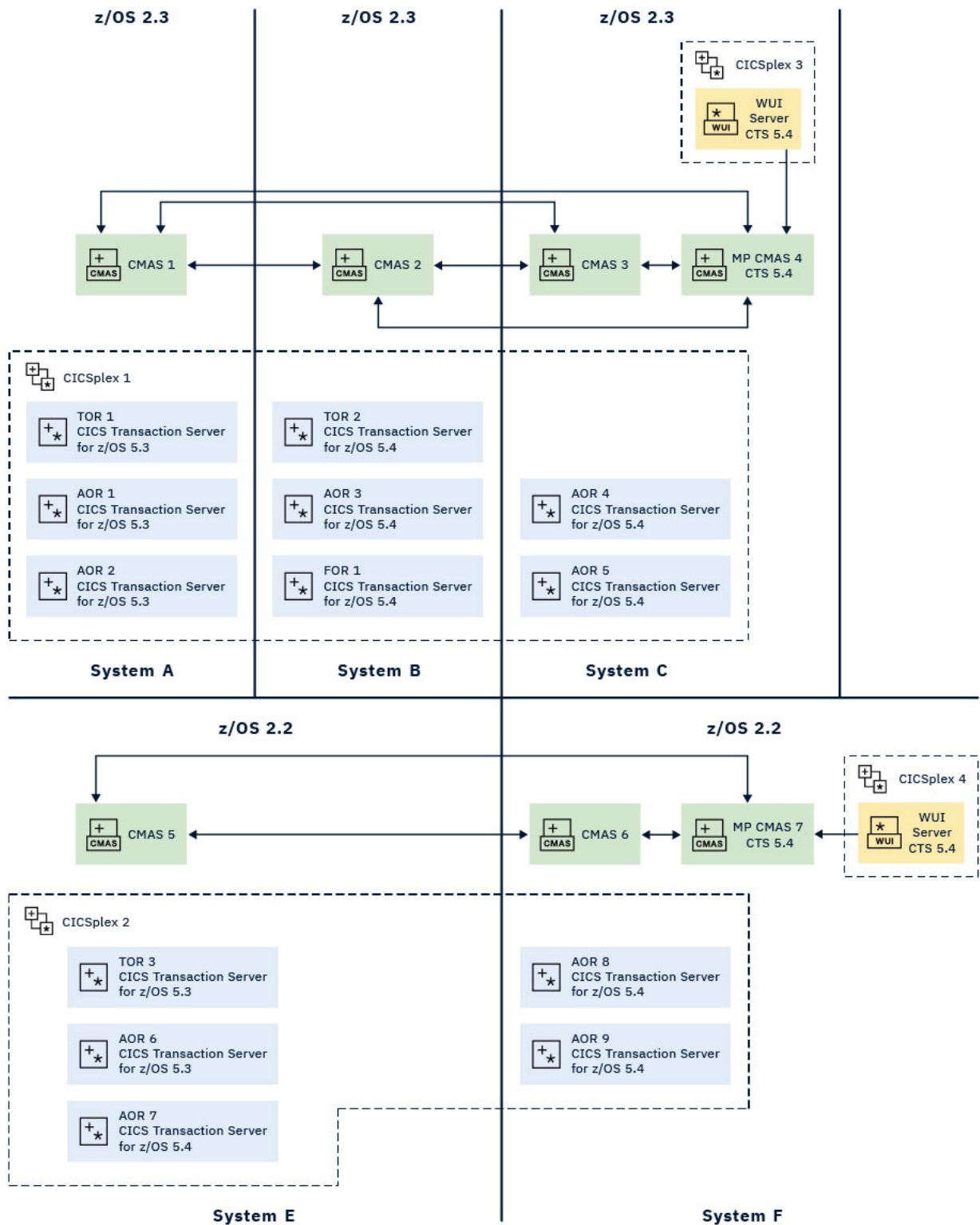


Figure 17. Adding WUI servers to the example map

## Naming your CICSplex SM entities

All the entities that you have defined, 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.

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 in the enterprise.
- Names of CICS systems and CICS system groups must be unique in the CICSplex.
- All other names must be unique for their type in the CICSplex.

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

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

Application resource definitions are:

- Db2<sup>®</sup> 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, LIBRARY definitions, map sets, sysplex enqueue models, TCP/IP services, temporary storage models

Region resource definitions are:

- Db2 entries, enterprise beans, journals, journal models, local shared resource (LSR) pools, profiles, temporary storage queue models, transaction classes, terminals, typeterms

System–system connectivity resource definitions are:

- Connections, Db2 connections, IPIC 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.

## 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 monitored once every 300 seconds. You do not need to name each transaction individually.

The rules governing the 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 CMAS-to-CMAS links would be much reduced in number (six CMAS-to-CMAS links, rather than 18), but the other decisions made about CICSplex 2 are still valid. That is, the same CICS system groups can be defined, and the CMAS installations are still required.





---

## Chapter 4. Planning CICSplex SM setup

CICSplex SM is installed as part of the product. To set up CICSplex SM, you must plan the security that is required, and which time zones to use in the CICSplex.

To implement CICSplex SM in your existing environment, you can use the samples that are supplied with CICS to get started.

---

### Security planning for CICSplex SM

CICSplex SM uses a SAF-compliant external security manager, such as RACF® to prevent unauthorized access to CICSplex SM functions and CICS resources, and 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.

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 RACF ID that has been used to sign on to CICSplex SM. Consider also the type of security checking you want to implement.

See [Implementing CICSplex SM security](#) for more information about how to set up CICSplex SM security.

#### Protecting access to CICSplex SM functions and CICS resources

To prevent unauthorized access, you create security profiles for combinations of CICSplex SM functions, and CICS resources that are to be protected. In most cases, the security provided by CICSplex SM security profiles is adequate.

An external security manager is also used to protect CICSplex SM libraries, procedures and Web User Interface resources. Full details of how to protect CICSplex SM libraries and procedures are provided in [Implementing CICSplex SM security](#). 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 [Controlling access to Web User Interface resources](#) for more information.

#### Special considerations for BAS

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.

Also take care if you use 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 other CICSplex SM security checking.

## Time zone planning for CICSplex SM

---

Much of the activity of CICSplex SM 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 **Time period definitions** 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 web 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. For a complete list of the codes, see [Time zone codes](#). 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. 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.

**Example:** Assume the following time zone settings in a CICSplex:

- The time-period definition time zone is S.
- The CMAS time zone is B.
- 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.

### What to do next

[Creating and managing time period definitions](#)

## 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 the CICSplex SM 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.



# Chapter 5. 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.

## Defining the CMAS configuration

For each CMAS you have defined, you must provide certain information to CICSplex SM.

The following information must be provided:

- 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

To enter this information, you can use the appropriate CMAS configuration administration views, accessed from the **CMAS configuration administration** menu, or the batched repository-update facility.

### WUI view sets and related objects for managing your CMAS configuration

The following WUI view sets and related objects are used to manage your CMAS configuration.

WUI view set	Object name	Description
Batched repository-update requests	BATCHREP	To create, update, remove, list, or dump one or more definitions from the data repository associated with the local CMAS.
CMAS to CMAS link definitions	CMTCMDEF	To display information about the direct LU6.2 and MRO communication links between the local CMAS and any other CMASs.
CICSplex definitions	CPLEXDEF	To define a CICSplex to the local CMAS and to administer CICSplex definitions for the local CMAS.
CMAS in CICSplex definitions	CPLXCMAS	To administer information about CMASs associated with a CICSplex, where the local CMAS is the maintenance point for that CICSplex.

### WUI view sets and resource objects for managing active CMASs

CICSplex SM also provides the following WUI view sets and resource objects that you can use to manage CMAS configuration definitions when the associated CMASs are active.

WUI view set	Object name	Description
CMASs managing CICSplex	CICSPLEX	To display information about the CMASs associated with a CICSplex known to the local CMAS.
CMASs known to local CMAS	CMAS	To display information about CMASs known to the local CMAS, shut down a CMAS, and alter CMAS component trace settings.

Table 3. View sets for managing active CMASs (continued)

WUI view set	Object name	Description
CICSplexes managed by CMAS	CMASPLEX	To display information about the CICSplexes being managed by the local CMAS and to remove the CMAS from managing CICSplexes when the MPSTATE is INVALID or NOTCONNECTED.
CMAS to CMAS links	CMTCLNK	To display information about all or specific CMASs linked to the local CMAS and to discard a CMAS-to-CMAS link.
CMAS to MAS links	CMTPLNK	To display information about all or specific MASs linked to the local CMAS and to discard the CMAS-to-MAS link.

## 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 26.

## 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 39. 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.

You identify a CICSplex identified to CICSplex SM with the **CICSplex definitions** view (CPLEXDEF object).

To enter this information, you can use either the appropriate **CMAS configuration administration** views, accessed from the **CMAS configuration administration** menu, 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 [“Planning CICS system definitions”](#) on page 48.

## Planning CICS system definitions

To manage a CICS region with CICSplex SM, you must associate the CICS region with a CICSplex that is defined to CICSplex SM. The definition that establishes this association also contains information about how the CICS system is to use the BAS, WLM, RTA, and monitoring components of CICSplex SM.

### About this task

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

### Procedure

1. Gather the basic information about your CICS region to create a definition.

a) Decide the name by which this CICS region is to be known to CICSplex SM.

This name can be up to 8 characters in length.

b) Find the APPLID, that is, the z/OS Communications Server application identifier of the CICS region.

c) Find the SYSID, that is, the CICS system identifier of the CICS region.

This information is sufficient to add the definition to the data repository. If you have special BAS, WLM, RTA, or monitoring requirements for any CICS region, you can gather the information before adding the definition or update the definition later.

2. Optional: Gather additional information about your CICS region for the definition.

a) If you are implementing system availability monitoring (SAM), find the name of the primary CMAS and the hours of operation of the CICS region.

The primary CMAS is the CMAS that the CICS region usually connects to.

b) Decide whether or not to simulate CICS command checking.

c) Decide whether or not to simulate CICS resource checking.

d) Decide what types of connections are required to other CICS regions and how many sessions to allow on each connection.

If you want to create IP interconnectivity (IPIC) connections, you must also the port number, the host name, and network ID of the CICS region.

e) Decide what time zone of the CICS region is to use, if it is not to be the same as that of its CMAS.

## What to do next

When you have gathered the information necessary to create a system definition, you can use the CICS Explorer or the **Topology administration** views in the Web User Interface to create the definition in the data repository. See [“Preparing CICS system group definitions” on page 49](#).

## 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 **Topology administration** (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 **System group definitions** view (CSYSGRP object).

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 **System group definitions** view (CSYSGRP object).

## CICS system definition views

The Web User Interface provides a number of views to help you define, install, and manage CICS system definitions. You can also use the CICS Explorer to perform the same tasks.

These views are described in detail in [CICSplex SM administration views](#).

Table 4. View sets to manage CICS system definitions

WUI view set	Object name	Description
ISC/MRO connection definitions	CONNDEF	To create and administer connection definitions that identify remote systems that a CICS region communicates with using intersystem communication (ISC) or multiple region operation (MRO).
CICS system definitions	CSYSDEF	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.
System group definitions	CSYSGRP	To create and administer CICS system groups within a CICSplex, including adding CICS regions to CICS system groups.
IPIC connection definitions	IPCONDEF	To create and administer TCP/IP connection definitions that describe the outbound attributes of connections between CICS regions that communicate using IP interconnectivity (IPIC) connections.
Time period definitions	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.
Session definitions	SESSDEF	To create and administer session definitions that describe the nature of logical connections between regions that communicate using intersystem communication (ISC) or multiple region operation (MRO).
CICS system links	SYSLINK	To create and administer the links between CICS regions in the CICSplex.
TCP/IP service definitions	TCPIPS	To create and administer TCP/IP connection definitions that describe the inbound attributes of connections between CICS regions.

CICSplex SM provides the **Runtime MAS display** (MAS) view set that you can use to manage configuration definitions when the associated CICS regions are active.



Table 5. View set to manage configuration definitions in an active CICS region

WUI view set	Object name	Description
Runtime MAS display	MAS	<p>To manage information about active CICS regions known to a CICSplex or connected to the designated CMAS involved in managing the CICSplex.</p> <ul style="list-style-type: none"> <li>• Display information about the active CICS regions</li> <li>• Browse or temporarily update the general, WLM, real-time analysis, resource monitoring, and MAS trace attributes of an active CICS region.</li> <li>• Stop MAS agent code in an active CICS region.</li> </ul>

**Note:** Using the **CICS system definitions** view (CSYSDEF object) and the **Time period definitions** view (PERIODEF object) to update definitions affects both a currently running system and the definitions in the data repository. Updates made using the **Runtime MAS display** view (MAS object), 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 [Managing resources using Business Application Services \(BAS\)](#), [Managing workloads through CICSplex SM](#), [Monitoring using real-time analysis \(RTA\)](#), or [Collecting statistics using CICSplex SM monitoring](#), as appropriate.



---

# Chapter 6. Administering resources with CICSplex SM

You can manage all of your CICS and CICSplex SM resources through any of the following interfaces; the CICS Explorer, the CICS Management Client Interface (CMCI), a WUI view, or using the CICSplex SM API.

Manage your resource definitions in the following ways:

- Using the CMCI, which accepts HTTP requests from your application.
- Using the CICS Explorer, which takes advantage of CMCI.
- Using the CICSplex SM Web User Interface (WUI).
- Adding CICSplex SM API commands directly to your application.

## Managing and installing resources

Use Business Application Services (BAS) to manage and install CICS resources that are stored on the CICSplex SM data repository.

Use CSD to manage and install CICS resources that are stored on the CICS system definition data set (CSD).

You must define a CSDGROUP for a CSD resource and a DEFVER for a BAS resource.

---

## Managing resources using Business Application Services (BAS)

Business Applications Services is the component of CICSplex SM that you use to manage the CICS resource definition and installation process for business applications at your enterprise.

### 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 runtime 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.

### BAS facilities

BAS enables you to view and manage your resources in terms of their use in your business, and independently of their physical location. It provides the following facilities:

#### Centralized resource definition

With BAS, you can implement resource definition and association across the entire CICSplex. Resource definitions are held in a CICS system definition data set (CSD). The CICSplex SM data repository (EYUDREP) can serve as the central repository for CICS resource definitions, accessed by all the CICS systems in the CICSplex. CICSplex SM 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 the attributes of a single definition.

- Managing multiple versions of a resource definition. For example, you can have different versions of the resources for an application as it progresses through a number of test phases and finally to production.

Each time you create a new definition for the same resource and the same name, you can assign a version number or let BAS assign an available version number. BAS can manage up to 15 versions of the same resource definition, so the version number is an integer in the range 1 through 15. Note that a new version number is not created if you update the resource. You can identify 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.

For more information, see [Multiple versions of a resource definition](#).

- Generating multiple CICS communication links from a single set of connection and session definitions.

You can create one set of BAS resource objects to define the connection definitions and reuse them in many CICS regions by using a system link (SYSLINK) object.

The system link definition describes the type of connection and connection definitions that are required to create a connection between a pair of CICS regions. You can use these connection definitions as a model to create any number of system links that share the same characteristics.

You can define different types of connections using SYSLINK objects:

- MRO or ISC connections require CONNDEF and SESSDEF resource objects to describe the CONNECTION and SESSION resources.
- IPIC connections require IPCONDEF and TCPDEF resource objects to describe the IPCONN and TCPIP SERVICE resources.

When you install a SYSLINK that uses these model definitions, the resource definitions are installed in the CICS regions automatically to create the connection.

For more information, see [CICS system link definitions](#).

### Logical scoping

When your CICS resources are defined to CICSplex SM, you can monitor and control those resources in terms of their participation in a named business application, rather than their physical location in the CICSplex. You can identify and refer to logically related resources as a set, regardless of where they reside at any given time.

You can reuse and associate sets of definitions 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.

### Distributed resource installation

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

## Supported resources

Business Application Services supports the following CICS resources:

### Application resources

These are the resources that support the business applications at your enterprise. They are the resources that an application requires to run:

- CICS BTS process types
- Db2 connections and transactions
- Document templates
- FEPI nodes, pools, property sets and targets
- Files and file key segment definitions
- IPIC connections
- LIBRARY resources
- Map sets
- Partition sets
- Pipeline
- Programs
- Sysplex enqueue models
- TCP/IP services
- Temporary storage models
- Transactions
- Transient data queues
- URI maps
- Web services

### **Region property resources**

These region property resources are the global resources that support the running of a CICS region:

- Journals
- Journal models
- Local shared resource (LSR) pools
- Profiles
- Transaction classes
- Terminals
- Typeterms

### **Connectivity resources**

These resources support the creation of connections between CICS regions and other systems:

- Connections
- Partners
- Sessions
- TCP/IP services
- IP interconnectivity (IPIC) connections

## **Two forms of BAS**

You can use BAS in two ways; the migration form, or in the full-function form. You can use either or both of these forms, depending on the situation and degree of precision you require.

- Migration form, in which resource groups are associated with resource descriptions. In this form, resource descriptions are similar to group lists. See [Figure 18 on page 56](#) and [“Using migration form BAS” on page 63](#).
- 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 19 on page 57](#) and [“Using full-function form BAS” on page 63](#).

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.

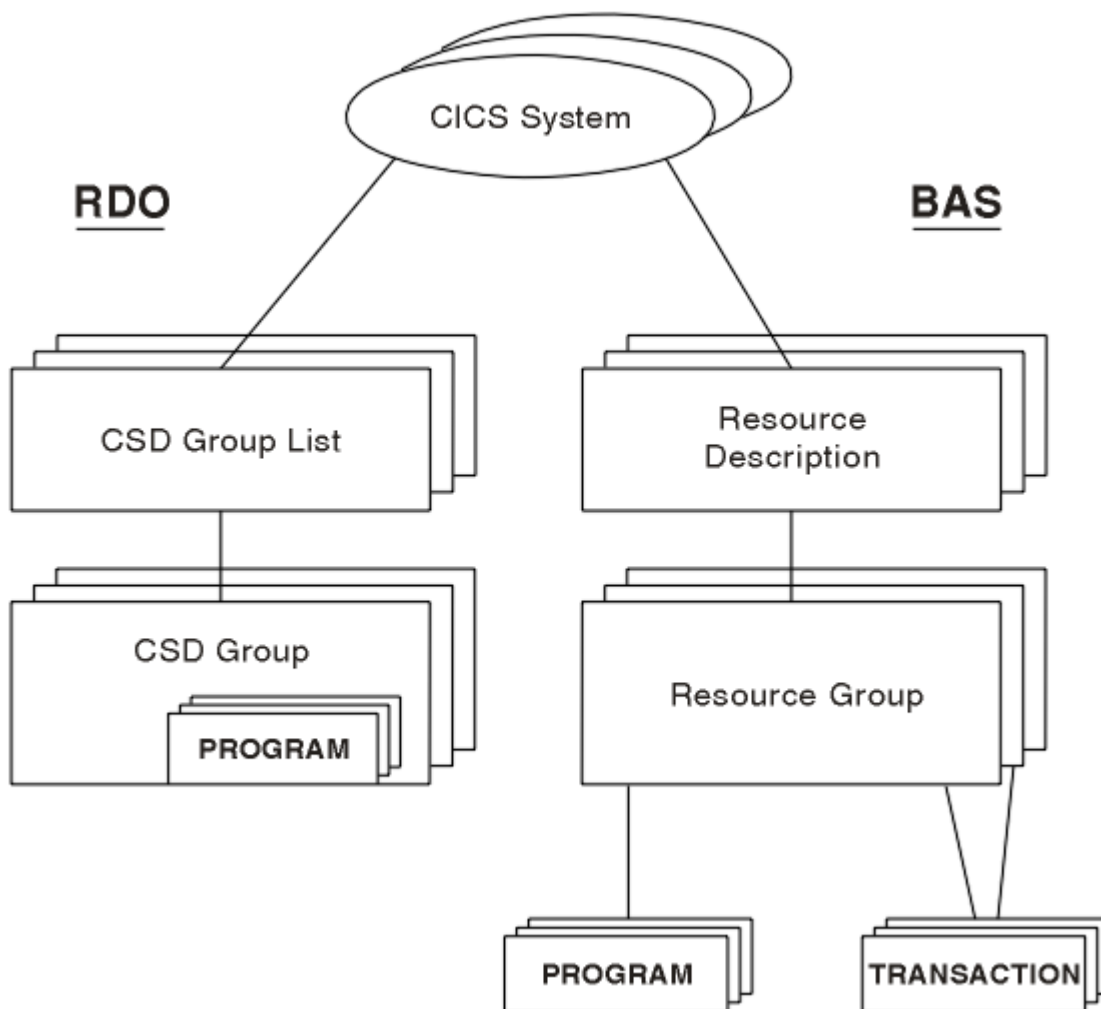


Figure 18. Migration from BAS

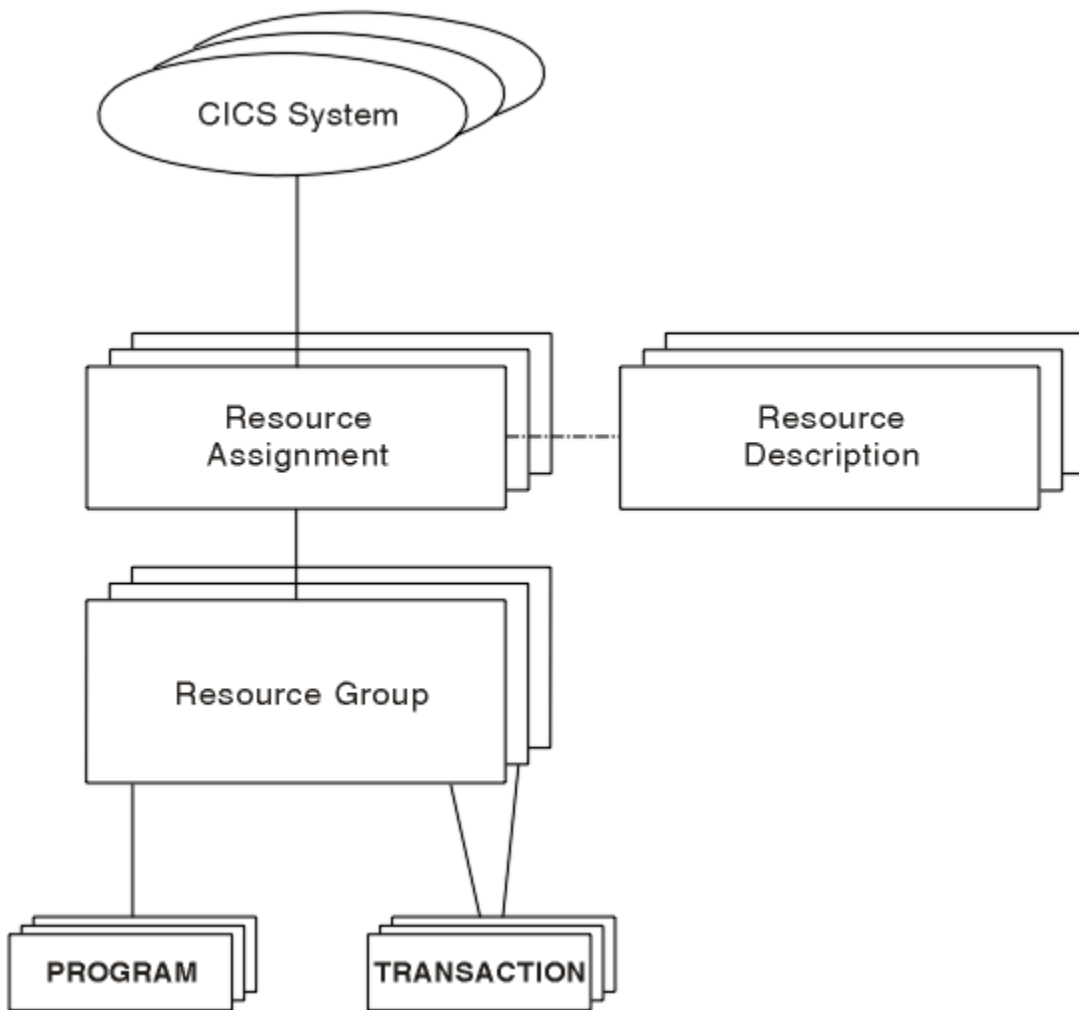


Figure 19. Full-function BAS

## Advantages of using BAS over RDO

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

## Limitations of RDO

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 CICS system definition data set (CSD) is shared.
- Each end of every communications link has to be defined explicitly.

## Advantages of BAS

BAS offers the following advantages:

- A familiar, RDO-like definition process with a choice of interfaces (WUI, 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 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 these WUI view sets and resource administration objects.

<i>Table 6. WUI view sets and resource administration objects</i>		
<b>WUI view set</b>	<b>Object name</b>	<b>Description</b>
Resource assignment definitions	RASGNDEF	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 <b>Resource description definitions</b> (RESDESC) view. 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.
Resource assignments in resource description	RASINDSC	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.
Resource selected by resource assignments	RASPROC	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.
Resource selected by resource description	RDSCPROC	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.
Resource description definitions	RESDESC	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.
Resource group definitions	RESGROUP	This object is used to associate one or more related resource definitions. The resource definitions may be of the same or different types.



<i>Table 6. WUI view sets and resource administration objects (continued)</i>		
<b>WUI view set</b>	<b>Object name</b>	<b>Description</b>
Resource groups in descriptions	RESINDSC	This view displays information about existing resource descriptions and the resource groups associated with them.
Resource definitions in resource groups	RESINGRP	This view displays information about existing resource groups and the resource definitions associated with them.
CICS system link definitions	SYSLINK	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.
CICS system resources	SYSRES	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.
Resource definitions	<i>resDEF</i> where <i>res</i> identifies the resource being defined	For each resource definition type, there is a resource definition object that defines the attributes for that definition. For the available resource types and the WUI views used to define them for CICSplex SM, see <a href="#">CICS resource definitions views</a> .

The object models showing the relationships between these objects are shown in [Figure 20 on page 60](#) and [Figure 21 on page 61](#). The view titles are followed by the resource name in parentheses.

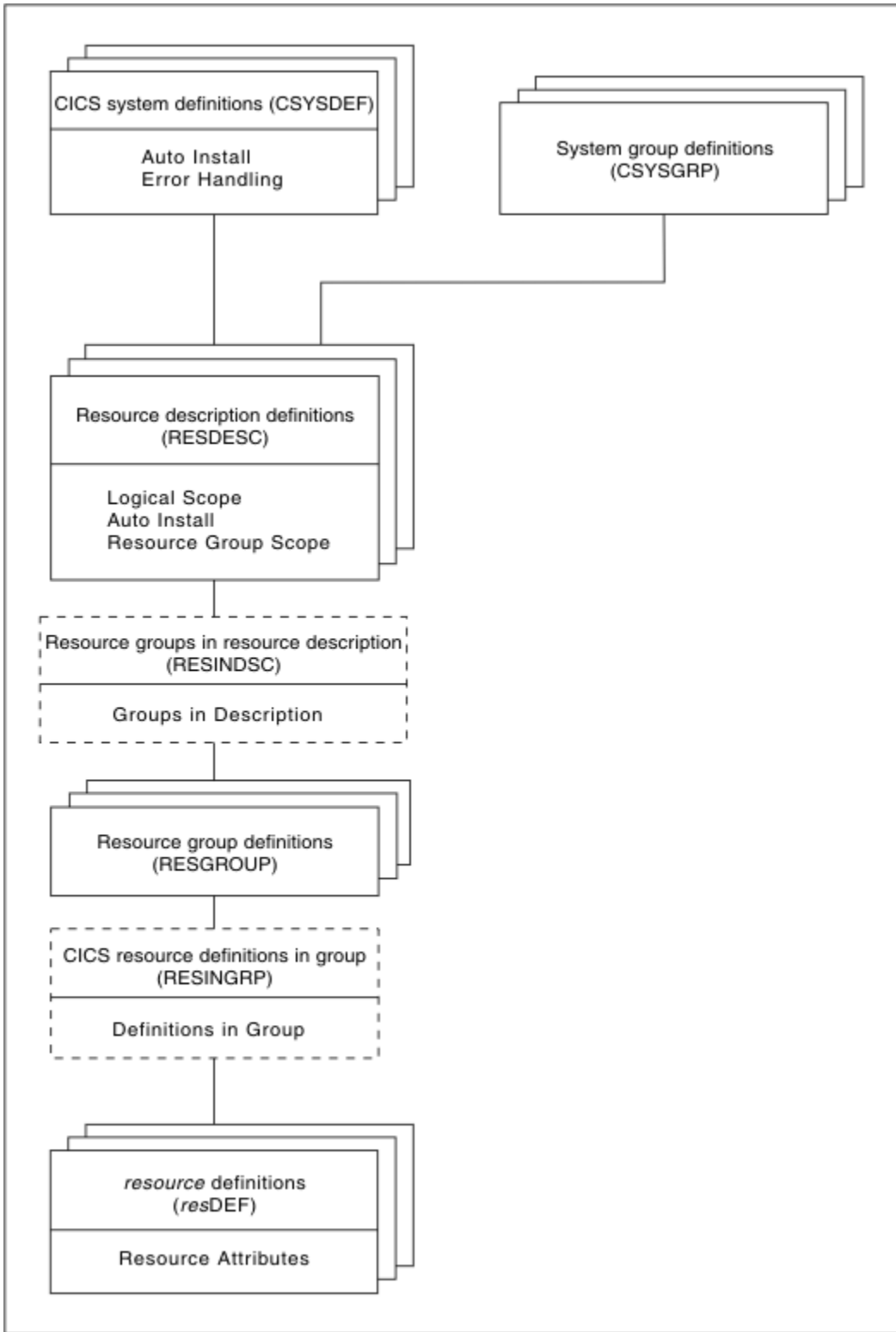


Figure 20. The migration from BAS object model

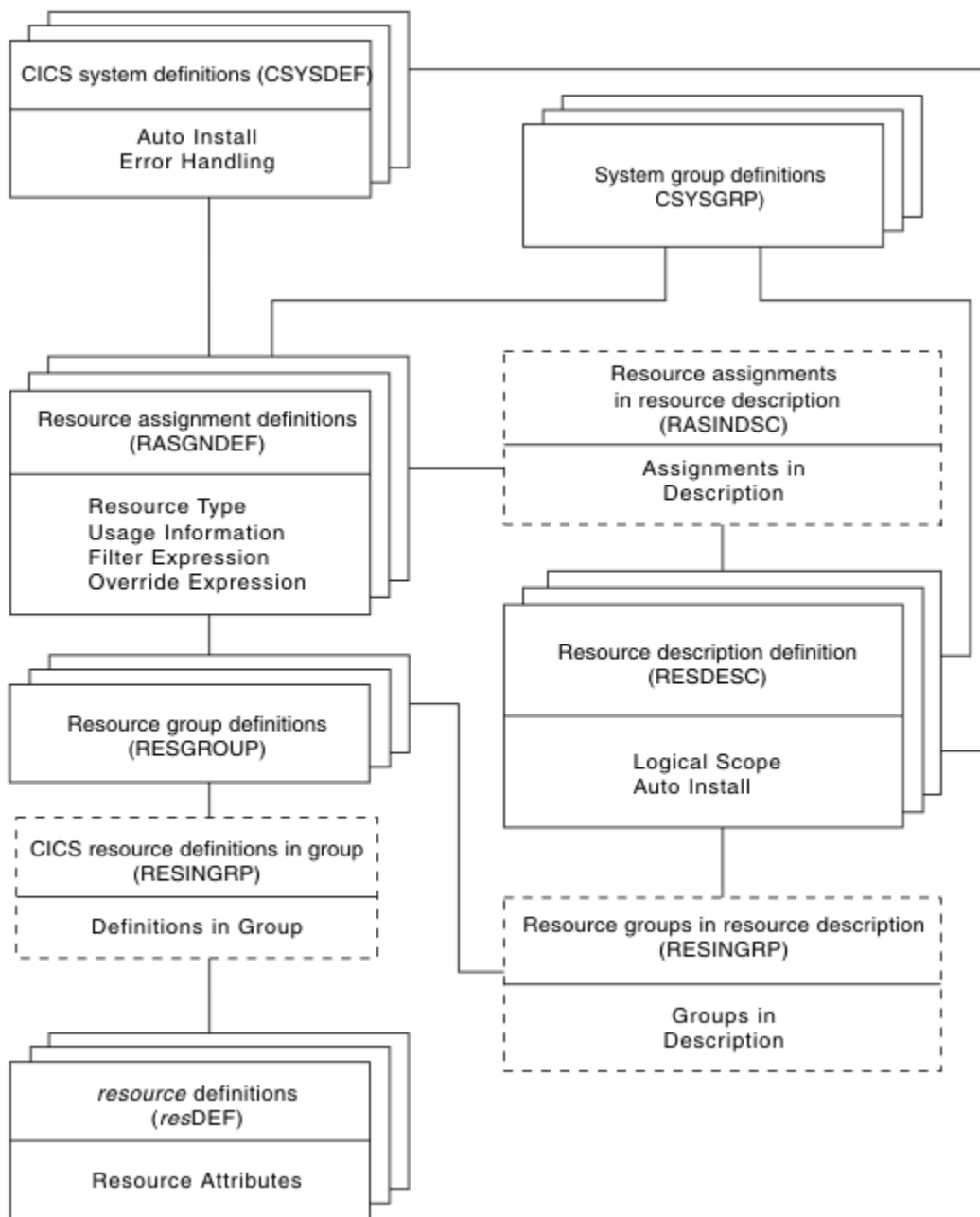


Figure 21. The full-function BAS object model

## Migrating your resource definitions

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

### About this task

To move your resource definitions and resource groups from your CSDs to the data repository, use the following process. 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.

### Procedure

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.

## Defining resources

You can use BAS to define and maintain resource definitions across a CICSplex. By creating resource definition objects in the data repository, you can use these objects as templates for creating large numbers of resource definitions.

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 have 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 resources. 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 region 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 region.
- You do not have 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” on page 62](#).

You can create resource definition objects in three ways:

- Use the administration views in CICS Explorer or the BAS administration and definition views in the Web User Interface.
- Use the batched repository-update facility; see [The batched repository-update facility](#).
- Use the CICSplex SM API; see [Developing CICSplex SM applications](#).

## 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.

## 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 **Resource group** view (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 20 on page 60](#).

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 form 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 21 on page 61](#).

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

## Using resource assignments

Power<sup>®</sup> 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 20 on page 60](#) and their functions are summarized in [Table 6 on page 58](#).

## Installing applications and resources

Optionally, as an alternative to installing resources from the CICS system definition data set (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 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 clicking the **install** button either from the appropriate resource definition view, or from one of the following views:

- **Resource group definitions** view.
- **Resource description definitions** view.
- **System link definitions** 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 **ISC/MRO connection definitions** views (CONNDEF object) and **Session definitions** views (SESSDEF object), 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 [BAS security considerations](#).

## 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 form BAS, and gradually move towards full-function BAS. You can use RDO, migration form 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 form 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?
- Which definitions are required before PLT processing? Definitions required before PLT phase 2 processing must be in the CSD.
- 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 65).

The CICS Interdependency Analyzer 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 CICS Interdependency Analyzer, see [CICS Interdependency Analyzer for z/OS Overview](#).

## Implementing BAS

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

### Creating migration for 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 CICS system definition data set (CSD) to the CICSplex SM data repository. This process, which is described in [“Migrating your resource definitions” on page 61](#), 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” on page 66](#).

### 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 **Resource assignment definitions** view (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 61](#). Alternatively:
  - Use the appropriate resource definition views to create resource definitions.
  - Use the **Resource group definitions** view to create resource groups (RESGROUP objects).
- Use the **Resource assignment definitions** view to create resource assignments (RASGNDEF object).
- Use the **Resource description definitions** view to create resource description to associate the definitions and assignments just created (RESDESC object).

Then, to create associations between these objects:



- Use the appropriate resource definition view to add resource definitions to a resource group (RESINGRP object).
- Use the **Resource group definitions** (RESGROUP) view to add the resource group to a resource description (RESINDESC object).
- Use the **Resource assignment definitions** view (RASGNDEF object) to associate the resource assignment with a resource description (RASINDSC object).
- Use the **Resource assignment definitions** view (RASGNDEF object) to associate the resource assignment with a CICS system or CICS system groups.

You can use the **Map** action button 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 **Resource definitions in resource groups** view (RESINGRP object).

## Where next?

Once you have identified your BAS requirements, you need to set up your resources and their associations.

This is described in [Creating resources with BAS](#).



---

## Chapter 7. Monitoring using real-time analysis (RTA)

Real-time analysis is the component of CICSplex SM that you use to define and manage notification of exception conditions at your enterprise.

When you have identified your RTA requirements, define them to CICSplex SM as described in [Working with real-time analysis administration views](#).

---

### What is CICSplex SM real-time analysis?

CICSplex SM 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.

The RTA functions are:

- [“System Availability Monitoring \(SAM\)” on page 70](#)
- RTA resource monitoring, which includes the following functions:
  - [“MAS resource monitoring \(MRM\)” on page 72](#)
  - [“Analysis point monitoring \(APM\)” on page 73](#)

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 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 **Local files** view set (LOCFILE object). 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.)

### Advantages of RTA

- 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.

## 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 operator messages that are written to the MVS console by default. External messages can also be viewed from the CICSplex SM **RTA outstanding events** view (EVENT object) while a condition is active.

When a condition has been resolved, it disappears from the **RTA outstanding events** view (EVENT object). (However, external messages written to the **RTA outstanding events** view (EVENT object) view are also written to the CICSplex SM EYULOG to provide a record of the contents of the **RTA outstanding events** view (EVENT object).) In the **RTA outstanding events** view (EVENT object), 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 all have the prefix "EYUPN". 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.

## System Availability Monitoring (SAM)

CICSplex SM has a system availability monitoring (SAM) function that 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:

#### **SAMMAX**

CICS has reached maximum tasks.

#### **SAMOPS**

The CICS system is not available.

#### **SAMNRM**

A CMAS has lost contact with an active MAS that it is managing.

The SAMNRM event is raised after the MAS fails to respond to a query from the CMAS for a period of 60 seconds. Unlike other SAM conditions, this event can be detected only when the CMAS interacts with the MAS. The condition is not detected if no CMAS to MAS interaction occurs.

#### **SAMSDM**

A CICS system dump is in progress.

#### **SAMSOS**

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

#### **SAMSTL**

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

#### **SAMTDM**

A CICS transaction dump is in progress.

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 **RTA outstanding events** view (EVENT object).

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 **RTA outstanding events** view (EVENT object).
- 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.

### **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 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 has an analysis point monitoring (APM) function which 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 WUI view sets and resource objects.

This is described in [Table 7 on page 73](#).

<b>WUI view set</b>	<b>Object name</b>	<b>Description</b>
Installed analysis definitions associated with an analysis point specification	APACTV	This object shows information about RTA definitions associated with an RTA point specification.
Action definitions	ACTION	An action definition designates the type of external notification that is to occur when the condition or conditions identified in an RTA definition are true.
RTA analysis point specifications	APSPEC	This object shows information about RTA point specifications. An RTA analysis point specification identifies one or more CMASs that are to be responsible for analyzing CICS systems within the CICSplex identified as the context.
RTA specifications in primary CMASs	CMDMPAPS	This object shows the association between RTA specifications and a primary CMAS.
RTA specifications in secondary CMASs	CMDMSAPS	This object shows the association between RTA specifications and secondary CMASs.
Evaluation definitions	EVALDEF	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 RTA definition is used to determine if a notifiable condition has occurred.
Real Time Analysis (RTA) outstanding events	EVENT	This object shows information about outstanding changes in the status of a CICSplex or one of its CICS systems.
Specifications to system group links	LNKSRSCG	This object shows CICS system groups associated with analysis specifications.
RTA specifications to CICS system links	LNKSRSCS	This object shows CICS systems associated with analysis specifications.

<i>Table 7. Real-time analysis objects (continued)</i>		
<b>WUI view set</b>	<b>Object name</b>	<b>Description</b>
Real Time Analysis (RTA) installed analysis and status definitions	RTAACTV	This object shows information about RTA and status definitions installed in CICS systems known to the CICSplex identified as the current context.
RTA definitions	RTADEF	An RTA definition identifies the evaluations to be performed on a periodic basis and the actions to be taken should a notifiable condition occur.
RTA groups	RTAGROUP	An RTA group is used to associate one or more related RTA definitions, status definitions, or both.
RTA group in analysis point specifications	RTAINAPS	This object displays the names of RTA groups associated with RTA point specifications.
RTA definitions in RTA groups	RTAINGRP	This object displays the names of RTA groups and the RTA and status definitions associated with them.
RTA groups in RTA specifications	RTAINSPC	This object shows the names of RTA specifications and the RTA groups associated with them.
RTA specifications	RTASPEC	An RTA specification identifies the default control attributes that are used for system availability monitoring and provides an anchor for all RTA definitions and status definitions associated with a CICS system.
Status definitions in RTA groups	STAINGRP	This object shows status definitions in RTA groups.
Status probe definitions	STATDEF	A status probe definition identifies a user-program that is to be called by CICSplex SM at specific intervals.

The RTA object model is shown in [Figure 22 on page 75](#). This figure shows the titles of the WUI views with the names of the resources in parentheses.



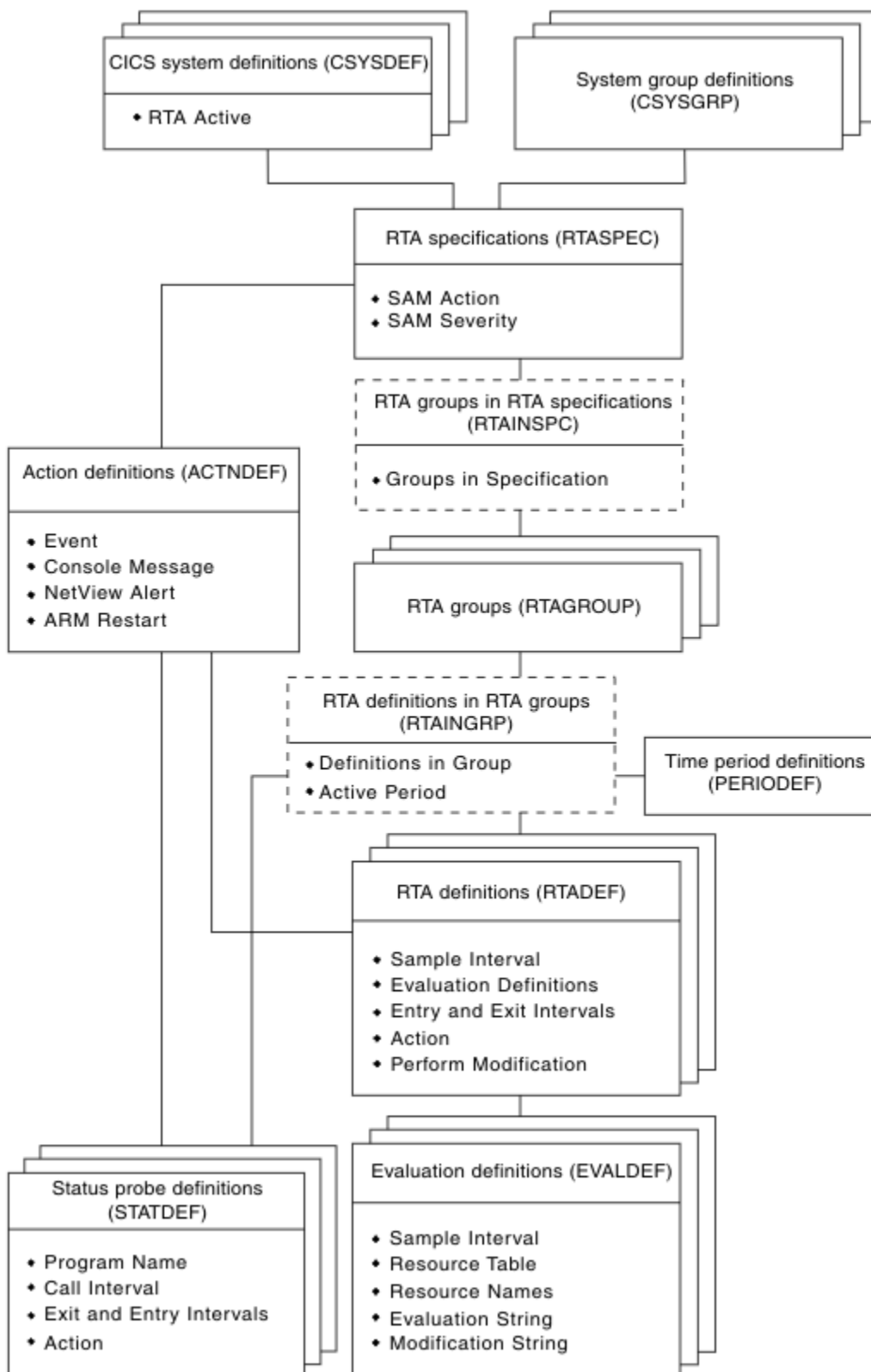


Figure 22. The real-time analysis object model

## 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 76) 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. Use the **Time period definitions** view to create time-period definitions (PERIODEF objects) 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 **RTA definitions** view to create an analysis definition (RTADEF object).
- Use the **Evaluation definitions** and **Action definitions** views to create the required subcomponents of an analysis definition (EVALDEF and ACTION objects).
- Use the **RTA groups** view to create an analysis group (RTAGROUP object).
- Use the **RTA specifications** view to create an analysis specification (RTASPEC object).

Then, to create associations between these objects:

- Use the **RTA definitions** view to add the analysis definition to the analysis group (RTAINGRP object).
- Use the **RTA groups** view to add the analysis group to the analysis specification (RTAINAPS object).

- Use the **RTA specifications to CICS system links** view (LNKSRSCS object) or the **RTA specifications to CICS system groups links** view (LNKSRSCG object) to associate the analysis specification with a CICS system or CICS system group. For more information about the LNKSxSCG parameters, see [Special attribute and resource table considerations: LNKSxSCG Records \(LNKSMSCG, LNKSRSCG, LNKSWSCG\)](#).

You can use the **Map** action button 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 **RTA definitions in RTA groups** view (RTAINGRP object), the **RTA groups in RTA specifications** view (RTAINSPC object), the **RTA specifications to CICS system links** view (LNKSRSCS object), and the **RTA specifications to CICS system groups links** view (LNKSRSCG object).

---

# Chapter 8. Collecting statistics using CICSplex SM monitoring

Monitoring is the component of CICSplex SM that you use to define and manage the collection of performance-related data in your enterprise.

## What is CICSplex SM 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.

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.

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** and **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 76), as these are monitored separately.

### CICSplex SM monitoring functions

CICSplex SM monitoring provides the following functions:

#### Monitoring of resources

CICSplex SM has monitor functions that can periodically monitor the performance of CICS resources.

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).

#### Monitoring 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, you specify a *monitor interval* for the CICSplex.

Use the **CICSplex definitions** view (CPLEXDEF object) to define a 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.

#### Sampling 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 CICSplex SM monitoring functions

The monitoring functions are set up and managed using the WUI view sets and resource objects.

These are described in [Table 8 on page 80](#).

<b>WUI view set</b>	<b>Object name</b>	<b>Description</b>
Monitor definitions	MONDEF	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.
Monitor groups	MONGROUP	A monitor group is used to associate one or more related monitor definitions.
Monitor definitions in groups	MONINGRP	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.
Monitor groups in monitor specifications	MONINSPC	This object displays the names of monitor specifications and the monitor groups associated with them.
Monitor specifications to system group links	LNKSMSCG	This object shows CICS system groups associated with monitor specifications.
Monitor specifications to CICS system links	LNKSMSCS	This object shows CICS systems associated with monitor specifications.
Active monitor specifications	POLMON	This object shows information about monitor definitions installed in CICS systems known to the CICSplex identified as the current context.

The monitoring object model is shown in [Figure 23 on page 81](#). The titles of the WUI views are followed by the resource names in parentheses.

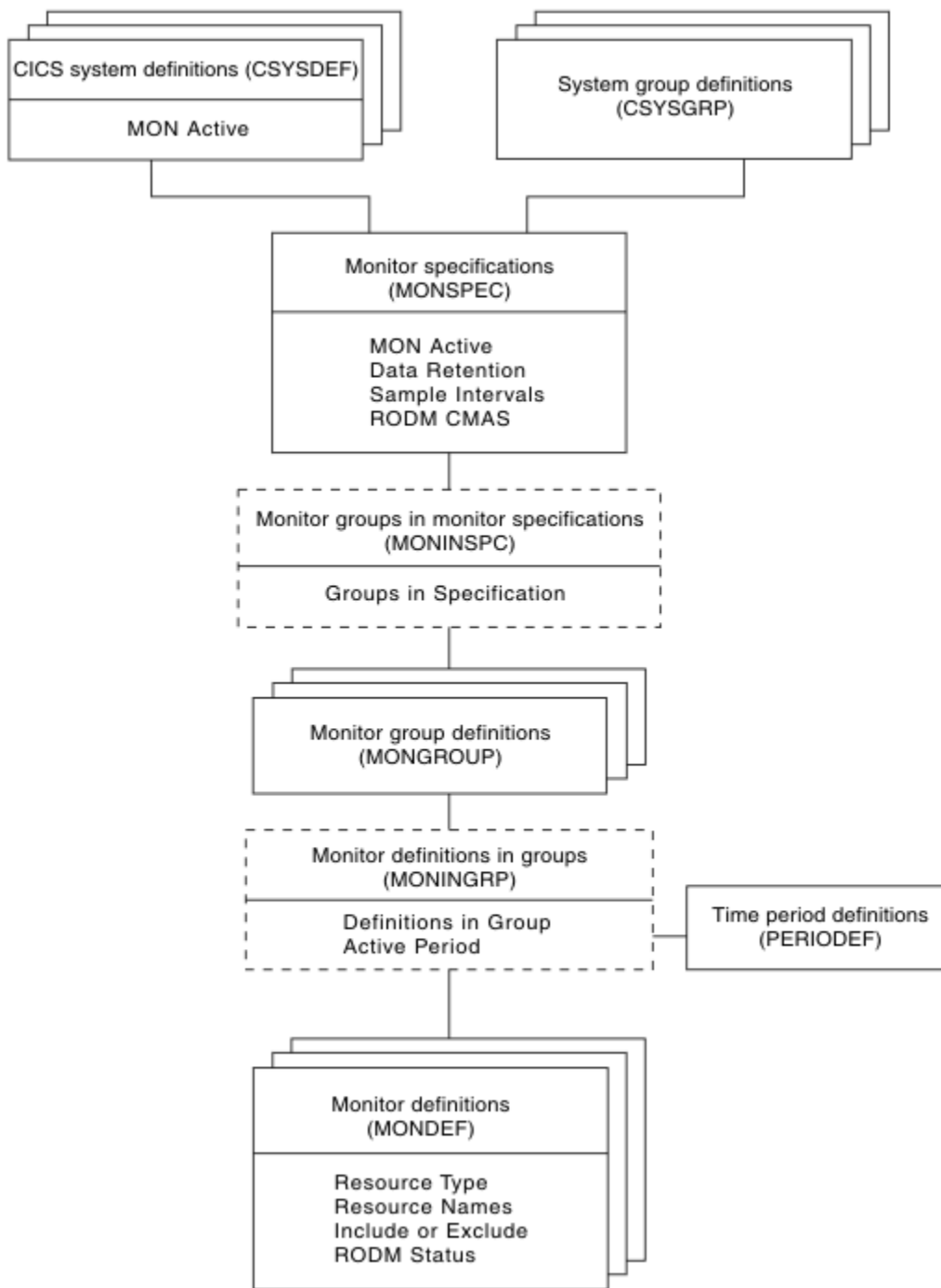


Figure 23. The resource monitoring object model

## Planning CICSplex SM monitoring

When you are planning what to monitor in your enterprise using CICSplex SM, the following recommendations can help you make best use of the monitoring functions that are available.

### 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-supplied resources? Often it is tempting to supply an asterisk (\*) as the resource name, which effectively means any instance of the specified resource. If you are not 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 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 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 **Runtime MAS display** view (MAS object), 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 do the following:

- 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 [Administering CICSplex SM](#) and [Resource monitoring](#).

For more information about the resource status facility see the description of the **RESSTATUS** parameter in [CICSplex SM system parameters](#).

## Using the CICS Monitoring Facility (CMF) with CICSplex SM

Some classes of monitoring data for CICSplex SM, in particular detailed task-related data that is displayed in some transaction monitoring 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 CICS region using the CICSplex SM operations functions, the CICS monitoring facility transaction CEMN, or the **SET MONITOR** command. Alternatively, you can activate performance class data monitoring permanently using CICS system initialization parameters.

CICS monitoring data is usually written to an SMF data set. However, if you prefer, you can collect monitoring data for use by CICSplex SM without having it written to an SMF data set, by specifying the CICSplex SM system parameter **SUPPRESSCMF**. For more information about this parameter, see [CICSplex SM system parameters](#).

For more information about the different classes of monitoring data, and about controlling the CICS monitoring facility, see [Introduction to CICS monitoring](#).

## A recommended approach

When you decide to implement monitoring functions in your enterprise, begin by defining the appropriate objects.

1. Create the following objects:

- Use the **Monitor definitions** view to create a monitor definition (MONDEF object).
- Use the **Monitor groups** view to create a monitor group (MONGROUP object).
- Use the **Monitor specifications** view to create a monitor specification (MONSPEC object).

2. Create associations between these objects:

- Use the **Monitor definitions** view to add the monitor definition to the analysis group (MONINGRP object).
- Use the **Monitor groups** view to add the monitor group to the analysis specification (MONINSPC object).
- Use the **Monitor specifications to CICS system links** view (LNKSMSCS object) and the **Monitor specifications to CICS system group links** view (LNKSMSCG object) to associate the monitor specification with a CICS system or CICS system group. For more information about the LNKSxSCG parameters, see [Special attribute and resource table considerations: LNKSxSCG Records \(LNKSMSCG, LNKSRS CG, LNKSWS CG\)](#).

You can use the **Map** button to display a visual map of the monitor definitions in your data repository. If you want to see a list of the objects and associations you have already defined, use the **Monitor definitions in groups** view (MONINGRP object), the **Monitor groups in monitor specifications** view (MONINSPC object), the **Monitor specifications to CICS system links** view (LNKSMSCS object), and the **Monitor specifications to CICS system group links** view (LNKSMSCG object) view.

## Where next?

---

Once you have identified your monitoring requirements, you should define them to CICSplex SM.

This is described in [Resource monitoring](#).

## Notices

---

This information was developed for products and services offered in the United States of America. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

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 rights 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 grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing  
IBM Corporation  
North Castle Drive, MD-NC119  
Armonk, NY 10504-1785  
United States of America*

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

*Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan Ltd.  
19-21, Nihonbashi-Hakozakicho, Chuo-ku  
Tokyo 103-8510, Japan*

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 jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore this statement may not apply to you.

This information 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.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who want 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 Director of Licensing  
IBM Corporation  
North Castle Drive, MD-NC119 Armonk,  
NY 10504-1785  
United States of America*

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 Client Relationship Agreement, IBM International Programming License Agreement, or any equivalent agreement between us.

The performance data discussed herein is presented as derived under specific operating conditions. Actual results may vary.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

#### COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

## Programming interface information

IBM CICS supplies some documentation that can be considered to be Programming Interfaces, and some documentation that cannot be considered to be a Programming Interface.

Programming Interfaces that allow the customer to write programs to obtain the services of CICS Transaction Server for z/OS, Version 5 Release 6 (CICS TS 5.6) are included in the following sections of the online product documentation:

- [Developing applications](#)
- [Developing system programs](#)
- [CICS TS security](#)
- [Developing for external interfaces](#)
- [Application development reference](#)
- [Reference: system programming](#)
- [Reference: connectivity](#)

Information that is NOT intended to be used as a Programming Interface of CICS TS 5.6, but that might be misconstrued as Programming Interfaces, is included in the following sections of the online product documentation:

- [Troubleshooting and support](#)
- [CICS TS diagnostics reference](#)

If you access the CICS documentation in manuals in PDF format, Programming Interfaces that allow the customer to write programs to obtain the services of CICS TS 5.6 are included in the following manuals:

- Application Programming Guide and Application Programming Reference
- Business Transaction Services

- Customization Guide
- C++ OO Class Libraries
- Debugging Tools Interfaces Reference
- Distributed Transaction Programming Guide
- External Interfaces Guide
- Front End Programming Interface Guide
- IMS Database Control Guide
- Installation Guide
- Security Guide
- CICS Transactions
- CICSplex System Manager (CICSplex SM) Managing Workloads
- CICSplex SM Managing Resource Usage
- CICSplex SM Application Programming Guide and Application Programming Reference
- Java Applications in CICS

If you access the CICS documentation in manuals in PDF format, information that is NOT intended to be used as a Programming Interface of CICS TS 5.6, but that might be misconstrued as Programming Interfaces, is included in the following manuals:

- Data Areas
- Diagnosis Reference
- Problem Determination Guide
- CICSplex SM Problem Determination Guide

## Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)<sup>®</sup> are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at [Copyright and trademark information at www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Apache, Apache Axis2, Apache Maven, Apache Ivy, the Apache Software Foundation (ASF) logo, and the ASF feather logo are trademarks of Apache Software Foundation.

Gradle and the Gradlephant logo are registered trademark of Gradle, Inc. and its subsidiaries in the United States and/or other countries.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

The registered trademark Linux<sup>®</sup> is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Red Hat<sup>®</sup>, and Hibernate<sup>®</sup> are trademarks or registered trademarks of Red Hat, Inc. or its subsidiaries in the United States and other countries.

Spring Boot is a trademark of Pivotal Software, Inc. in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Zowe™, the Zowe logo and the Open Mainframe Project™ are trademarks of The Linux Foundation.

The Stack Exchange name and logos are trademarks of Stack Exchange Inc.

## Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

### Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

### Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

### Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

### Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

## IBM online privacy statement

IBM Software products, including software as a service solutions, (*Software Offerings*) may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information (PII) is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect PII. If this Software Offering uses cookies to collect PII, specific information about this offering's use of cookies is set forth below:

### For the CICSplex SM Web User Interface (main interface):

Depending upon the configurations deployed, this Software Offering may use session and persistent cookies that collect each user's user name and other PII for purposes of session management, authentication, enhanced user usability, or other usage tracking or functional purposes. These cookies cannot be disabled.

### For the CICSplex SM Web User Interface (data interface):

Depending upon the configurations deployed, this Software Offering may use session cookies that collect each user's user name and other PII for purposes of session management, authentication, or other usage tracking or functional purposes. These cookies cannot be disabled.

### For the CICSplex SM Web User Interface ("hello world" page):

Depending upon the configurations deployed, this Software Offering may use session cookies that do not collect PII. These cookies cannot be disabled.

**For CICS Explorer:**

Depending upon the configurations deployed, this Software Offering may use session and persistent preferences that collect each user's user name and password, for purposes of session management, authentication, and single sign-on configuration. These preferences cannot be disabled, although storing a user's password on disk in encrypted form can only be enabled by the user's explicit action to check a check box during sign-on.

If the configurations deployed for this Software Offering provide you, as customer, the ability to collect PII from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see [IBM Privacy Policy](#) and [IBM Online Privacy Statement](#), the section entitled *Cookies, Web Beacons and Other Technologies* and the [IBM Software Products and Software-as-a-Service Privacy Statement](#).





---

# Index

## Special Characters

, introduction to [1](#)

## A

action buttons [2](#)  
Action definitions view [73](#)  
Active monitor specifications [80](#)  
address space (CMAS)  
  definitions [47](#)  
  links to other CMASs [33](#)  
  maintenance point [37](#)  
  NetView [8](#)  
  performance considerations [32](#)  
  use of ESSS [8](#)  
  where to install [32](#)  
ADMCONFIG menu [48](#)  
administering the topology of a CICSplex [48](#)  
agent code, MAS [7](#)  
aggregation  
  group [13](#)  
analysis definition [74](#)  
analysis group [74](#)  
analysis point monitoring (APM)  
  automation and [69](#)  
  description of [73](#)  
  implementing [77](#)  
analysis specification [74](#)  
AOC CICS Automation [70](#)  
application programming interface (API) [3](#)  
automatic installation  
  of a monitor definition [83](#)  
  of an analysis definition [77](#)  
  of resources [64](#)

## B

batched repository-update facility  
  description of [4](#)  
  EXTRACT routine [66](#)  
Business Application Services (BAS)  
  creating resource definitions [62](#)  
  definition of [2](#)  
  description of [53](#)  
  EXTRACT routine [66](#)  
  functions [53](#)  
  objects [9](#)  
  resource installation [53](#)

## C

CICS Monitoring Facility (CMF) [84](#)  
CICS system [26](#)  
CICS system group  
  defining [49](#)

CICS system group (*continued*)  
  identifying [49](#)  
  using instead of CICSplexes [29](#)  
CICS system objects [26](#)  
CICS system resources view [59](#)  
CICSplex  
  definition [6](#)  
  definitions [48](#)  
  identifying [28](#)  
  when to have more than one [29](#)  
CICSplex definitions view [48](#)  
CICSplex SM address space (CMAS)  
  release level [8](#)  
CICSplex topology  
  administering [48](#)  
  defining [48](#)  
CMAS [8](#)  
CMAS configuration administration view [47](#)  
Configuration administration view [48](#)  
CPLEXDEF object [48](#)  
creating resource definitions [62](#)  
CSYSGRP object [49](#)

## D

data repository [4, 9](#)  
data-repository initialization job [44](#)  
defining the CMAS configuration [47](#)  
defining the topology of a CICSplex [48](#)  
dynamic installation of resources [64](#)  
dynamic routing [2](#)

## E

entity types  
  CICSplex SM [39](#)  
Environment Services System Services (ESSS) [8](#)  
Evaluation definitions view [73](#)  
external messages, RTA [70](#)  
external notifications, RTA [70](#)  
EXTRACT routine [66](#)  
EYU9DXDUT [44](#)

## F

functions of [1](#)

## G

generic alert, SNA  
  CMAS considerations [32](#)  
generic names  
  CICSplex SM [39](#)  
globalization support [37](#)  
graphql  
  api [13](#)

group, CICS system [31](#), [49](#), [70](#)

## I

IBM CICS Explorer [4](#)  
inconsistent set [62](#)  
installation of resources  
    automatic [64](#)  
    dynamic [64](#)  
installation planning [43](#)  
Installed analysis definitions associated with an analysis  
point specification view [73](#)

## L

language support [37](#)  
links, communication  
    CMAS-to-CMAS [33](#)  
LNKSMSCG object [80](#)  
LNKSMSCS object [80](#)  
local MAS [7](#)

## M

maintenance point CMAS [37](#)  
managed application system (MAS)  
    definition [7](#)  
map of CICS systems, producing [26](#)  
MAS agent code [7](#)  
MAS resource monitoring (MRM)  
    automation and [72](#)  
    description of [72](#)  
    implementing [77](#)  
menu  
    ADMCONFIG [48](#)  
    CMAS configuration administration menu [47](#)  
    Topology administration [48](#)  
MONDEF object [80](#)  
MONGROUP object [80](#)  
MONINGRP object [80](#)  
MONINSPC object [80](#)  
monitor definition [80](#)  
Monitor definitions in groups view [80](#)  
Monitor definitions view [80](#)  
monitor group [80](#)  
Monitor groups in monitor specifications view [80](#)  
Monitor groups view [80](#)  
monitor interval [82](#)  
Monitor specifications to CICS system links [80](#)  
Monitor specifications to system group links [80](#)  
monitoring  
    CICS Monitoring Facility (CMF) [84](#)  
    definition of [3](#)  
    introduction to [79](#)  
    monitor interval [82](#)  
    sample interval [82](#)

## N

naming convention  
    entities [39](#)  
    Starter Set [39](#)  
NetView [8](#), [32](#)

notifications, external, RTA [70](#)

## O

object  
    ACTDEF [73](#)  
    APACTV [73](#)  
    APSPEC [73](#)  
    CMDMPAPS [73](#)  
    CMDMSAPS [73](#)  
    CPLEXDEF [48](#)  
    CSYSGRP [49](#)  
    CSYSGRP object [49](#)  
    EVALDEF [73](#)  
    LNKSMSCG [80](#)  
    LNKSMSCS [80](#)  
    LNKSRSCG [73](#)  
    LNKSRSCS [73](#)  
    MAS [50](#)  
    MONDEF [80](#)  
    MONGROUP [80](#)  
    MONINGRP [80](#)  
    MONINSPC [80](#)  
    PERIODEF [44](#), [76](#)  
    POLMON [80](#)  
    RASGNDEF [58](#)  
    RASINDSC [58](#)  
    RASPROC [58](#)  
    RDSCPROC [58](#)  
    RESDESC [58](#)  
    RESGROUP [58](#), [63](#)  
    RESINDSC [59](#)  
    RESINGRP [59](#)  
    RTAACTV [74](#)  
    RTADEF [74](#)  
    RTAGROUP [74](#)  
    RTAINAPS [74](#)  
    RTAINGRP [74](#)  
    RTAINSPC [74](#)  
    RTASPEC [74](#)  
    STAINGRP [74](#)  
    STATDEF [74](#)  
    SYSLINK [59](#)  
    SYSRES [59](#)  
object model, recommended usage  
    business application services [66](#)  
    monitoring [84](#)  
    real-time analysis [77](#)  
operating objects [9](#)  
operations views [2](#)

## P

performance considerations  
    CMAS [32](#)  
    CMAS-to-CMAS links [33](#)  
    of monitoring [81](#)  
PERIODEF object [44](#)  
planning WUI servers [37](#)  
POLMON object [80](#)

## R

- RASGNDEF object [58](#)
- RASINDSC object [58](#)
- RASPROC object [58](#)
- RDSCPROC object [58](#)
- real-time analysis (RTA)
  - analysis point monitoring (APM) [73](#), [77](#)
  - AOC CICS Automation [70](#)
  - definition of [3](#)
  - external messages [70](#)
  - implementing [76](#)
  - introduction to [69](#)
  - MAS resource monitoring (MRM) [72](#), [77](#)
  - planning [76](#)
  - SNA generic alert [70](#)
  - system availability monitoring (SAM) [70](#), [76](#)
- RESDESC object [58](#)
- RESGROUP object [58](#)
- RESINDSC [59](#)
- RESINGRP object [59](#)
- resource assignment
  - migrating toward using [66](#)
- Resource assignment process view [58](#)
- Resource assignment view [58](#)
- Resource assignments in resource description view [58](#)
- resource checking [62](#)
- Resource definition in resource group view [59](#)
- Resource description process view [58](#)
- Resource descriptions view [58](#)
- Resource group (RESGROUP) view [63](#)
- Resource groups in description view [59](#)
- Resource groups view [58](#)
- Resource Object Data Manager (RODM)
  - activating the interface to [83](#)
- resource status facility
  - activating [83](#)
- resources
  - automatic installation of [64](#)
  - dynamic installation of [64](#)
- retention period
  - monitoring [82](#)
- RTA analysis point specifications view [73](#)
- RTA definitions in RTA groups view [74](#)
- RTA definitions view [74](#)
- RTA group in analysis point specifications view [74](#)
- RTA groups in RTA specifications view [74](#)
- RTA groups view [74](#)
- RTA installed analysis and status definitions view [74](#)
- RTA specifications in primary CMASs view [73](#)
- RTA specifications in secondary CMASs view [73](#)
- RTA specifications to CICS system links view [73](#)
- Runtime MAS display (MAS) view [50](#)

## S

- sample interval
  - monitoring [82](#)
- security
  - BAS [65](#)
  - planning [43](#)
- single point of control
  - definition of [2](#)
- single-system image (SSI)

- single-system image (SSI) (*continued*)
  - definition of [1](#)
  - role of CMAS in providing [8](#)
- SNA generic alert
  - CMAS considerations [8](#), [32](#)
- Specifications to system group links view [73](#)
- Specifications view [74](#)
- staged implementation [41](#)
- STATDEF view [74](#)
- Status definitions in RTA groups view [74](#)
- SYSLINK object [59](#)
- SYSRES object [59](#)
- system availability monitoring (SAM)
  - automation and [71](#)
  - customizing external notifications [71](#)
  - description of [70](#)
  - implementing [76](#)
- system group [31](#)
- System group definition view [49](#)
- System link view [59](#)

## T

- time period definitions view [44](#)
- Time period definitions view [76](#)
- time zone [4](#), [44](#)
- topology
  - administration [48](#)
- Topology administration menu [48](#)

## V

- validating resources [62](#)
- view
  - CMAS configuration administration [47](#)
  - Time period definitions [44](#)

## W

- Web User Interface [4](#)
- Web User Interface (WUI)
  - globalization support [37](#)
  - where to locate [37](#)
- workload management (WLM)
  - definition of [2](#)





