

CICS Transaction Server for z/OS



# Installation Guide

*Version 3 Release 2*



CICS Transaction Server for z/OS



# Installation Guide

*Version 3 Release 2*

**Note!**

Before using this information and the product it supports, be sure to read the general information under "Notices" on page 427.

This edition applies to Version 3 Release 2 of CICS Transaction Server for z/OS, program number 5655-M15, and to all subsequent versions, releases, and modifications until otherwise indicated in new editions.

© Copyright IBM Corporation 1989, 2011.

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

---

# Contents

<b>Preface</b> . . . . .	xiii
What this book is about . . . . .	xiii
Who is this book for? . . . . .	xiii
What you need to know to understand this book . . . . .	xiii
How to use this book . . . . .	xiv
for planning . . . . .	xiv
for installation . . . . .	xiv
Notes on terminology. . . . .	xv
CICS system connectivity . . . . .	xv
<b>Summary of changes</b> . . . . .	xvii
Changes for CICS Transaction Server for z/OS, Version 3 Release 2 . . . . .	xvii
Changes for CICS Transaction Server for z/OS, Version 3 Release 1 . . . . .	xvii
Changes for CICS Transaction Server for z/OS Version 2 Release 3. . . . .	xvii
Changes for CICS Transaction Server for z/OS Version 2 Release 2. . . . .	xviii
Changes for CICS Transaction Server for z/OS, Version 2 Release 1 . . . . .	xviii
Changes for CICS Transaction Server for OS/390 Version 1 Release 3. . . . .	xix
CICSplex SM installation and setup . . . . .	xix

---

## Part 1. Planning for Installation . . . . . 1

<b>Chapter 1. Planning for installation</b> . . . . .	3
Introduction . . . . .	3
Overview of CICS TS . . . . .	3
Packaging . . . . .	3
CICS TS elements and features . . . . .	4
Exclusive and non-exclusive elements and features. . . . .	4
What you receive with CICS TS . . . . .	4
Documentation . . . . .	6
The CICS Clients and The CICS Transaction Gateway. . . . .	6
<b>Chapter 2. Requirements for CICS TS</b> . . . . .	7
Hardware requirements for installing CICS TS . . . . .	7
Software requirements for installing CICS TS . . . . .	7
z/OS UNIX file system and PDSE requirements . . . . .	8
DASD storage requirements for CICS TS . . . . .	8
<b>Chapter 3. Planning for CICSplex SM</b> . . . . .	9
CICSplex SM setup checklist and worksheets . . . . .	9
MVS installation and setup checklist . . . . .	9
System worksheet. . . . .	11
CMAS worksheet . . . . .	12
MAS worksheet. . . . .	13
WUI server worksheet . . . . .	13
IVP planning worksheet. . . . .	14
Planning for CICSplex SM migration . . . . .	15

---

## Part 2. Installation . . . . . 17

<b>Chapter 4. Installing CICS TS</b> . . . . .	19
Installing CBPDO . . . . .	19
Using the SMP/E RECEIVE, APPLY, and ACCEPT commands . . . . .	20

Using the DFHISTAR process . . . . .	20
Installing ServerPac . . . . .	22
What you need to install the ServerPac . . . . .	22
Summary of the ServerPac installation steps . . . . .	23
The next step . . . . .	23
<b>Chapter 5. Installing CICS TS using DFHISTAR . . . . .</b>	<b>25</b>
Edit the DFHISTAR Job. . . . .	25
Specify the CICS Transaction Server temporary installation libraries . . . . .	31
Specify the JOB parameters for installation jobs. . . . .	31
Specify the scope of the installation . . . . .	32
Specify the type of JES to be used . . . . .	32
Specify the utilities to be used . . . . .	32
Specify the prefix of CICS Transaction Server jobs. . . . .	33
Specify the indexes of CICS Transaction Server data sets . . . . .	33
Specify the CICS TS z/OS UNIX directories and data sets . . . . .	34
Specify extra qualifiers . . . . .	35
Specify block sizes . . . . .	35
Specify the disk unit for work data sets . . . . .	35
Specify SMS option for DASD allocations . . . . .	35
Specify disk volumes. . . . .	36
Allocate space for CICS Transaction Server disk volumes . . . . .	39
Specify Attributes of the Temporary SMP/E Work Data Sets . . . . .	39
Specify Attributes of the Permanent SMP/E Data Sets . . . . .	40
Specify SMP/E Zone Attributes . . . . .	41
Specify the High-level Qualifiers for SMP/E Data Sets . . . . .	44
Specify the distribution tape device type . . . . .	44
Specify attributes of the CICS Transaction Server system data sets . . . . .	44
Specify Attributes of any additional target Libraries. . . . .	45
Specify the Data Set Name of the SISLOAD Library . . . . .	46
Specify the Data Set Name of the CSSLIB Library . . . . .	46
Specify the Data Set Name of the SCEECPP Library . . . . .	46
Specify the Data Set Name of the SCEELKED Library . . . . .	47
Specify the Data Set Name of the SCEELKEX Library . . . . .	47
Specify the Data Set Name of the SCEEOBJ Library . . . . .	47
Specify the Data Set Name of the SCEESAMP Library . . . . .	47
Specify the Data Set Name of the SCLBSID Library . . . . .	47
Specify the Data Set Name of the SCSQLOAD Library . . . . .	47
Specify the Data Set Name of the SCSQANLE Library . . . . .	48
Specify the Data Set Name of the SCSQCICS Library . . . . .	48
Specify the Data Set Name of the SCSQAUTH Library . . . . .	48
Specify the installation directory for Java support . . . . .	48
Specify the Data Set Name of the SDSNLOAD Library . . . . .	48
Specify the Data Set Name of the SEZARPCL and SEZACMTX libraries . . . . .	49
Specify the Data Set Name of the SCEECICS & SCEERUN libraries . . . . .	49
Specify the Data Set Name of the SCEERUN2 & SCEELIB libraries . . . . .	49
Specify the Data Set Name of the SCSFMOD0 and SIXMEXP libraries . . . . .	49
Specify the Data Set Name of the SCEEBND2 library . . . . .	50
Specify Log stream and Log stream Structure Attributes. . . . .	50
Specify attributes specific to CICSplex SM. . . . .	50
Create RACF Profiles for the CICS Transaction Server Data Sets . . . . .	53
Run the DFHISTAR Job . . . . .	53
Check the Output from the DFHISTAR Job . . . . .	54
Check that You Are Ready to Run the Installation Jobs . . . . .	55
Run the Installation Jobs . . . . .	56
Run Times of the Installation Jobs . . . . .	56

The DFHIHFS0 Job . . . . .	56
The DFHIHFS1 Job . . . . .	57
The DFHISMKD Job . . . . .	58
The DFHINST1 Job . . . . .	58
The DFHINST2 Job . . . . .	58
The DFHINST3 Job . . . . .	58
The DFHINST4 Job . . . . .	59
The DFHINST5 Job . . . . .	59
The DFHINST6 Job . . . . .	60
The DFHIJVMJ Job . . . . .	61
Check the Output from the Installation Jobs . . . . .	61
What next? . . . . .	61
Copy the CICS Transaction Server procedures into a procedure library . . . . .	62
Create Extra Sets of CICS Transaction Server Target Libraries (Optional) . . . . .	64
Load the CICS Transaction Server Source Material (Optional) . . . . .	66
Load Other Optional Features . . . . .	66
Activating CICS Transaction Server . . . . .	66
Checklist for the CICS Transaction Server for z/OS Installation . . . . .	67
<b>Chapter 6. Installing the CICS Information Center . . . . .</b>	<b>73</b>
Requirements . . . . .	73
Installing the information center on a workstation . . . . .	73
Installing the information center on a server . . . . .	74
Installing the CICS plug-in in an Eclipse IDE or help system . . . . .	75

---

## **Part 3. Setting up the MVS environment for CICS . . . . . 77**

<b>Chapter 7. Authorizing the CICS and CICSplex SM libraries . . . . .</b>	<b>79</b>
<b>Chapter 8. Authorizing CICS regions to access MVS resources . . . . .</b>	<b>81</b>
Protecting CICS load module data sets . . . . .	81
Authorizing access to data set services modules . . . . .	82
Authorizing access to a CICS region's VTAM ACB . . . . .	82
Authorizing the region userid to access category 1 transactions . . . . .	83
Authorizing access to an SMSVSAM server . . . . .	83
Authorizing access to MVS log streams . . . . .	84
Authorizations for users of IXCMIAPU . . . . .	84
Authorizations for CICS regions . . . . .	85
Giving CICS regions access to z/OS UNIX System Services . . . . .	86
Activating RACF resource classes . . . . .	89
<b>Chapter 9. Defining the default CICS userid to RACF . . . . .</b>	<b>91</b>
<b>Chapter 10. Installing CICS-required modules in the MVS linklist . . . . .</b>	<b>93</b>
CICS- and CICSplex SM-supplied modules required in the MVS linklist . . . . .	93
CICS shared data tables modules for the MVS linklist . . . . .	95
CICSplex SM API linklist modules . . . . .	95
Installing the REXX function package . . . . .	96
Modules of other MVS products in the MVS linklist . . . . .	97
<b>Chapter 11. Defining CICS as an MVS subsystem . . . . .</b>	<b>99</b>
The IEASYSxx MVS initialization member . . . . .	99
Noting IEASYSxx values for CICSplex SM . . . . .	100
Updating IEASYSxx (CMAS) . . . . .	101
The IEFSSNaa MVS subsystem initialization member . . . . .	101
The SYS1.PARMLIB(BPXPRMxx) parameters . . . . .	102

	The console message-handling facility . . . . .	103
	The DFHSSly message-formatting initialization member . . . . .	104
	Default message-formatting initialization parameters . . . . .	105
	Activating message formatting . . . . .	106
	Modules needed to use the console message-handling facilities . . . . .	107
	Coexistence considerations . . . . .	107
I	EXCI pipe allocation . . . . .	108
I	The DFHSSly initialization member . . . . .	108
I	The EXCI pipe allocation limit . . . . .	109
I	Default EXCI pipe allocation limit initialization parameter . . . . .	109
	<b>Chapter 12. Installing the CICS Type 3 SVC . . . . .</b>	<b>111</b>
	Defining the CICS SVCs to your MVS . . . . .	111
	Using more than one version of the CICS Type 3 SVC . . . . .	112
	MRO between different CICS releases with a changed SVC number . . . . .	113
	<b>Chapter 13. Selecting the high-performance option . . . . .</b>	<b>115</b>
	Defining DFHPSVC to MVS . . . . .	115
	Loading module DFHPSVC . . . . .	115
	Removing existing DFHPSVC modules from the MVS nucleus . . . . .	116
	<b>Chapter 14. Defining CICS regions as applications to VTAM . . . . .</b>	<b>117</b>
	Defining specific CICS APPL statements to VTAM . . . . .	117
	VTAM APPL parameters for CICS regions . . . . .	117
	VTAM version and release level indicator . . . . .	119
	Message DFH3473 on opening the VTAM ACB . . . . .	119
	Cross-domain considerations . . . . .	119
	VTAM LU alias facility . . . . .	120
	Defining VTAM requirements for CICSplex SM . . . . .	124
	Step 1: Creating a VTAM application definition for a CMAS . . . . .	125
	Step 2: Defining cross-domain resources for a CMAS . . . . .	125
	Step 3: Updating the configuration list for a CMAS . . . . .	126
	Step 4: Activating the major nodes for a CMAS . . . . .	126
	<b>Chapter 15. Installing CICS modules in the MVS link pack area . . . . .</b>	<b>129</b>
	Preparing to install CICS modules in the MVS link pack area . . . . .	129
	Benefits of using the MVS link pack area . . . . .	129
	What is meant by the MVS link pack area? . . . . .	129
	Which modules must be in the MVS link pack area? . . . . .	130
	Which modules can be in the MVS link pack area? . . . . .	131
	Service considerations . . . . .	132
	How to use modules in the MVS link pack area . . . . .	133
	Space requirements for CICS modules in the MVS link pack area . . . . .	133
	Defining the CICS LPA library to your MVS . . . . .	133
	Installing CICS modules in the LPA . . . . .	134
	Selecting modules for the MVS link pack area . . . . .	134
	The CICS-supplied SMP/E USERMOD, DFHUMOD . . . . .	135
	Changing a CICS-supplied USERMOD . . . . .	135
	Preparing the USERMOD . . . . .	135
	Receiving and applying the USERMOD . . . . .	135
	Controlling the use of modules from the MVS link pack area . . . . .	136
	Modules in the MVS link pack area from hlq.SDFHAUTH . . . . .	136
	Modules in the MVS link pack area from hlq.SDFHLOAD . . . . .	136
	Sample DFHCSDUP job to specify USELPACOPY(YES) . . . . .	140
	<b>Chapter 16. Installing CICSplex SM modules in the MVS link pack area . . . . .</b>	<b>141</b>



Space requirements . . . . .	141
Installing CICSplex SM modules into the LPA . . . . .	141
Receiving and applying the USERMOD . . . . .	142
Defining the SYS1.CICSTS32.CPSM.SEYULPA library to your MVS . . . . .	143
Applying maintenance to LPA modules. . . . .	143
<b>Chapter 17. Defining CICS IPCS exit control data to MVS . . . . .</b>	<b>145</b>
The DFHIPCSP CICS exit control data . . . . .	145
<b>Chapter 18. Preparing to use the CICSplex SM IPCS tools . . . . .</b>	<b>147</b>
Updating BLSCECT . . . . .	147
Updating library allocations . . . . .	148
SDUMP options . . . . .	148
<b>Chapter 19. MVS Program properties table entries . . . . .</b>	<b>149</b>
RACF password checking . . . . .	149
Non-swappable CICS regions . . . . .	149
MVS protection key for CICS . . . . .	149
<b>Chapter 20. MVS performance definitions . . . . .</b>	<b>151</b>
Implementing MVS workload management . . . . .	151
Matching CICS performance parameters to service policies . . . . .	152
<b>Chapter 21. MVS automatic restart management definitions . . . . .</b>	<b>153</b>
Implementing MVS automatic restart management . . . . .	153
<b>Chapter 22. MVS cross-system MRO definitions . . . . .</b>	<b>155</b>
Sysplex overview . . . . .	155
MVS XCF considerations for MRO . . . . .	155
<b>Chapter 23. PR/SM policy for handling MVS failures . . . . .</b>	<b>157</b>
XCF PR/SM policy . . . . .	157
<b>Chapter 24. MVS ASREXIT - SYMREC Authorization Exit . . . . .</b>	<b>159</b>
<b>Chapter 25. Definitions required for VSAM RLS support . . . . .</b>	<b>161</b>
Defining the master coupling facility lock structure . . . . .	161
Defining coupling facility cache structures and cache sets . . . . .	161
Defining cache sets. . . . .	162
Defining SMS storage classes for RLS access. . . . .	162
Altering data set attributes for RLS access . . . . .	162
Defining sharing control data sets . . . . .	163
Authorizing CICS userids for VSAM RLS support. . . . .	164
Adding new parameters to SYS1.PARMLIB(IGDSMSxx) . . . . .	164
Establishing new procedures for VSAM RLS support . . . . .	164
Activating the coupling facility structures . . . . .	165
<b>Chapter 26. Console messages. . . . .</b>	<b>167</b>
<b>Chapter 27. Defining the logger environment for CICS journaling . . . . .</b>	<b>169</b>
Requirements planning and checklist . . . . .	169
Setting up the environment for CICS log manager . . . . .	174
Coupling facility or DASD-only? . . . . .	176
Coupling facility log streams . . . . .	177
Defining coupling facility structures . . . . .	178
Planning considerations for the number of log structures . . . . .	181

Log structure naming conventions . . . . .	182
Defining coupling facility log streams . . . . .	182
Sizing considerations for coupling facility log streams . . . . .	184
Coupling facility requirements in an RLS environment . . . . .	193
Staging data sets for coupling facility log streams. . . . .	194
DASD-only log streams . . . . .	195
Defining DASD-only log streams . . . . .	196
Sizing considerations for DASD-only log streams . . . . .	197
Converting a DASD-only log stream to use a coupling facility . . . . .	200
Analyzing SMF Type 88 records . . . . .	201
Managing secondary storage . . . . .	202
Managing log data sets . . . . .	202
Log tail management . . . . .	203
<b>Chapter 28. Enabling Unicode data conversion by z/OS . . . . .</b>	<b>205</b>
Selecting conversion images . . . . .	205
<b>Chapter 29. Applying service to CICS Transaction Server for z/OS . . . . .</b>	<b>207</b>
Load library secondary extents . . . . .	207
The CICS TS-supplied SMP/E procedure. . . . .	207
APAR fixes . . . . .	208
PTFs . . . . .	208
CICS service considerations . . . . .	209
Servicing the CICS messages data set . . . . .	209
<b>Chapter 30. Applying service to CICSplex SM . . . . .</b>	<b>211</b>
CICS Transaction Server for z/OS-supplied SMP/E procedure . . . . .	211
Applying PTFs to CICSplexes running CICS Transaction Server for z/OS, Version 3 Release 2 and later . . . . .	211
CICSplex SM service considerations . . . . .	212

---

## **Part 4. Getting ready to run CICS . . . . . 213**

<b>Chapter 31. Getting ready to run CICS . . . . .</b>	<b>215</b>
Tailoring the CICS-supplied skeleton jobs. . . . .	215
The CICS installation libraries . . . . .	215
What you should do . . . . .	216
Running DFHISTAR . . . . .	216
CICSplex SM post-installation members . . . . .	217
Creating the CICS data sets . . . . .	218
Naming conventions . . . . .	219
Creating data sets common to all CICS regions, DFHCOMDS job. . . . .	220
Creating data sets unique to each CICS region, DFHDEFDS job . . . . .	221
Creating the CICS messages data set, DFHCMACI job . . . . .	223
Defining the sample applications data sets . . . . .	223
CICS resource definitions for CICSplex SM . . . . .	226
Dynamic creation of CICS resource definitions. . . . .	227
Overriding the dynamically created CICS resource definitions for CICSplex SM . . . . .	227
Journalmodel considerations in a CMAS . . . . .	229
Defining DL/I support . . . . .	230
PDIRs. . . . .	230
Adding remote DL/I support. . . . .	230
Global user exits for DL/I. . . . .	232
Adding CICS support for programming languages . . . . .	232
Installing Language Environment support. . . . .	233

Verifying your Java components installation . . . . .	236
Verifying Java components checklist . . . . .	236
Authorizing the hlq.SDFJAUTH library . . . . .	238
Amending parameters in SYS1.PARMLIB(BPXPRMxx) . . . . .	238
Installing MRO and ISC support . . . . .	239
Installing MRO support . . . . .	239
Adding communications support between systems . . . . .	241
Enabling TCP/IP in a CICS region . . . . .	243
Using TCP/IP in a CICS region . . . . .	244
Enabling REXX for CICS . . . . .	245
Step 1--Modify your RDO definitions to add required entries . . . . .	245
Step 2--Create the RFS filepools . . . . .	245
Step 3--BIND the CICSQL program to your DB2 plan . . . . .	246
Step 4--Add DD statements to your CICS startup job . . . . .	246
Step 5--Modify member CICSTART . . . . .	246
Step 6--Format the RFS filepools . . . . .	246
Step 7--Verifying the installation . . . . .	247
Step 8--Accessing the supplied softcopy documentation . . . . .	247
Customization information for REXX . . . . .	247
Setting up a CICSplex SM address space (CMAS) . . . . .	252
Before you begin . . . . .	252
CICSplex SM auxiliary storage usage . . . . .	253
Preparing to transmit generic alerts to NetView . . . . .	254
Preparing to start a CMAS . . . . .	254
Setting the VTAM APPLID for a CMAS . . . . .	270
CMAS journaling . . . . .	270
Shutting down a CMAS . . . . .	273
Restarting a CMAS . . . . .	274
Setting up a CICS managed application system (MAS) . . . . .	274
Before you begin . . . . .	274
Using CICS global user exits and user-replaceable modules . . . . .	275
Controlling the use of modules from the LPA . . . . .	275
Preparing a MVS MAS . . . . .	276
Stopping and restarting management of a CICS system . . . . .	283
Setting up a CICSplex SM Web User Interface server . . . . .	284
Preparing a CICS system to act as the Web User Interface server . . . . .	284
Configure CICS Web support . . . . .	286
Review CICS system initialization parameters . . . . .	286
Specify language and codepage . . . . .	287
Prepare codepage conversion table . . . . .	288
Create the Web User Interface server repository (EYUWREP) . . . . .	288
WUI data set creation and customization . . . . .	290
Specify the customizable view and menu help data set . . . . .	292
Specify the Web User Interface server initialization parameters . . . . .	292
Transient data queue definitions . . . . .	301
Temporary storage models . . . . .	301
Specify the JCL DD statements . . . . .	301
Security considerations . . . . .	302
Additional tasks . . . . .	304
CICSplex SM system parameters . . . . .	305

---

**Part 5. CICS verification . . . . . 321**

<b>Chapter 32. CICS verification . . . . .</b>	<b>323</b>
Running the installation verification procedures . . . . .	323
Preparation for running the IVPs . . . . .	323

Resources for the CICS messages facility, CMAC . . . . .	328
The CICS startup procedure, DFHSTART . . . . .	328
Verify batch job, DFHIVPBT. . . . .	330
Verify interactive job, DFHIVPOL . . . . .	335
Verifying shared data tables support . . . . .	345
Verifying the CICS-DBCTL interface. . . . .	351
Testing the CICS-DB2 environment . . . . .	356
Running the EJB "Hello World" sample . . . . .	358

---

**Part 6. CICSplex SM verification . . . . . 359**

<b>Chapter 33. CICSplex SM verification . . . . .</b>	<b>361</b>
Using DFHISTAR to verify CICSplex SM has installed correctly . . . . .	361
Checking the MVS environment . . . . .	362
Generating samples using the DFHISTAR job . . . . .	362
Defining the data sets . . . . .	363
Creating a CICSplex SM address space (CMAS). . . . .	363
Creating a CICSplex SM Web User Interface server . . . . .	364
Testing your Web User Interface . . . . .	365
Using the Web User Interface to define a CICSplex and a MAS . . . . .	366
Creating a CICSplex SM managed CICS system (MAS) . . . . .	367
Verifying IVP is complete using the WUI . . . . .	367
Shutting down the MAS, WUI server, and CMAS . . . . .	368
Setting up a more complex configuration . . . . .	368

---

**Part 7. Appendixes . . . . . 369**

<b>Appendix A. Information about modules eligible for the MVS link pack area . . . . .</b>	<b>371</b>
Priority . . . . .	371
Size . . . . .	372
Option/Note . . . . .	372
<b>Appendix B. CICSplex SM modules eligible for the MVS link pack area . . . . .</b>	<b>393</b>
<b>Appendix C. Default CICS resource definitions for CICSplex SM. . . . .</b>	<b>395</b>
<b>Bibliography . . . . .</b>	<b>403</b>
The CICS Transaction Server for z/OS library . . . . .	403
The entitlement set . . . . .	403
PDF-only books . . . . .	403
Other CICS books . . . . .	405
Books from related libraries . . . . .	405
Systems Network Architecture (SNA) . . . . .	405
Advanced communications function for VTAM (ACF/VTAM) . . . . .	405
NetView Version 3.1 . . . . .	406
NetView MultiSystem Manager Version 2.2 . . . . .	406
DATABASE 2 (DB2) . . . . .	406
eNetwork Communications Server for OS/2 Warp, Version 5 . . . . .	406
Virtual Storage Access Method (VSAM) . . . . .	406
Resource Access Control Facility (RACF). . . . .	406
System Modification Program Extended (SMP/E) . . . . .	406
Sysplex planning. . . . .	407
DFSMS/MVS . . . . .	407
MVS . . . . .	407

Determining if a publication is current . . . . .	407
<b>Accessibility</b> . . . . .	409
<b>Index</b> . . . . .	411
<b>Notices</b> . . . . .	427
<b>Trademarks</b> . . . . .	429



---

# Preface

---

## What this book is about

This book describes CICS® Transaction Server for z/OS®, Version 3 Release 2 (CICS TS). It takes you through the necessary planning and helps you install CICS Transaction Server for z/OS, Version 3 Release 2. It contains guidance about tailoring CICS for use in your systems environment. It:

- Describes the content of CICS TS and the two delivery methods—ServerPac and CBPDO
- Explains the method of installing CICS TS (with either ServerPac or CBPDO), and provides references to the required sources of information. In this book, “installing” means loading the code into the libraries in preparation for the migration and customizing tasks.
- Lists the hardware and software you must have to run the CICS TS product elements and exploit the function provided by CICS TS.
- Covers installation, and verification of that installation, for both CICS and CICSplex® SM.
- Tells you about installing the workstation-based components of CICS TS.

The book assumes that you are upgrading to CICS TS levels of all the product elements in the Server.

Planning the migration to CICS TS requires that you understand the function provided by the CICS TS product set. You can learn about the function in the various product libraries of the individual elements that comprise the product set.

Always check the product libraries (for example, in the the *CICS Transaction Server for z/OS Migration from CICS TS Version 2.3* manual) for changes that might affect CICS TS elements.

---

## Who is this book for?

This book is intended for experienced CICS system programmers who are planning to migrate to CICS TS.

This book is also for system programmers who are responsible for installing and tailoring CICS and CICSplex SM.

By “experienced”, we mean that a system programmer’s experience includes installing and managing CICS and some or all of the other elements in the CICS TS product set.

---

## What you need to know to understand this book

To fully understand the installation information in this book, you should have experience of the IBM® MVS™ operating system, and the System Modification Program/Extended (SMP/E) licensed program needed to maintain CICS and CICSplex SM. To use the installation verification procedures, you should be familiar with the JCL and cataloged procedures for MVS. It also helps if you are familiar with CICS and CICSplex SM concepts.

---

## How to use this book

### for planning

Read through the sections of this book that tell you about:

- The products and hardware you need to support the function that comes with CICS TS.
- The pointers to migration and installation information that is in the product libraries that you receive with CICS TS.

Once you have identified the actions you need to take to complete your migration, write a plan describing the steps your installation will take. Include your estimate of how much time each step requires and who will do it.

### for installation

CICS and CICSplex SM are available only as elements of the CICS Transaction Server, through either the ServerPac or CBPDO method of delivery. For information about these two methods of delivery of the CICS Transaction Server, see Chapter 4, “Installing CICS TS,” on page 19.

To install the CICS Transaction Server using the CBPDO method, you should use the *CICS Transaction Server for z/OS Program Directory*, together with the instructions contained in the Memo to Users Extension, to load the software from the tape DASD. For the ServerPac method, you follow the supplied set of ISPF dialogs and the accompanying documentation.

After you have loaded the CICS Transaction Server elements to DASD, you should then use this book to tailor CICS to your environment; that is to:

- Integrate CICS with MVS and ACF/VTAM
- Apply service to CICS (if required)
- Create the CICS data sets
- Install DB2<sup>®</sup> support (if required)
- Install MRO and ISC support (if required)
- Run the installation verification procedures (if required).

**Note:**

1. Table 28 on page 375 gives details of the CICS modules that are needed in, and eligible for, the MVS link pack area.
2. If you installed CICS from CBPDO, you do not need to run the DFHISTAR job again to specify the post-installation parameters. However, if you wish to create several copies of the post-installation jobs (for example to create several copies of the DFHDEFDS job to define CICS data sets unique to several CICS regions), you can edit and run the DFHISTAR job as many times as required.

Some of the information in this book is also of interest if you have installed CICS Transaction Server using the ServerPac method of delivery.

In particular, you should edit and run the DFHISTAR job, specifying the keyword POST, to define parameters needed to tailor your CICS environment.



---

## Notes on terminology

**CICS** is used throughout this book to mean the CICS element of the IBM CICS Transaction Server for z/OS, Version 3 Release 2.

**CICSplex SM** refers to CICSplex System Manager, an element of CICS Transaction Server.

**CICS TS Version 2 region** is used to refer to a CICS region running under CICS TS Version 2.

**MVS** is used throughout this book to mean the operating system MVS, or the Base Control Program (BCP) element of z/OS.

**RACF**<sup>®</sup> is used throughout this book to mean the MVS Resource Access Control Facility (RACF) or any other external security manager that provides equivalent function.

The term **CICS TS 3.2 region** is used to refer to a CICS region running under CICS Transaction Server for z/OS, Version 3 Release 2.

**\$** In the programming examples in this book, the dollar symbol (\$,) is used as a national currency symbol and is assumed to be assigned the EBCDIC code point X'5B'. In some countries a different currency symbol, for example the pound symbol (£), or the yen symbol (¥), is assigned the same EBCDIC code point. In these countries, the appropriate currency symbol should be used instead of the dollar symbol.

*hlq* Throughout this book, the term *hlq* is used to denote the high-level qualifier of the CICS TS data sets; for example, CICSTS32.CICS for CICS data sets and CICSTS32.CPSM for CICSplex SM data sets. The CICSTS32 part of the high-level qualifier is defined by the LINDEX parameter in the DFHISTAR installation job.

### IMS<sup>™</sup> library names

The IMS libraries referred to in this section are identified by IMS.libnam (for example, IMS.SDFSRESL). If you are using your own naming conventions, change the IMS prefix to match those naming conventions.

---

## CICS system connectivity

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

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

- CICS Transaction Server for z/OS, Version 3 Release 2
- CICS Transaction Server for z/OS, Version 3 Release 1
- CICS Transaction Server for z/OS, Version 2 Release 3
- CICS Transaction Server for z/OS, Version 2 Release 2

You can use this release of CICSplex SM to control systems running supported releases of CICS that are connected to, and managed by, your previous release of CICSplex SM. However, if you have any directly-connectable release levels of CICS, as listed above, that are connected to a previous release of CICSplex SM, you are strongly recommended to migrate them to the current release of CICSplex

SM, to take full advantage of the enhanced management services. See the relevant *CICS Transaction Server for z/OS Migration Guide* for information on how to do this.

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

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

<b>CICS system</b>	<b>CICSplex SM component of CICS TS 3.2</b>	<b>CICSplex SM component of CICS TS 3.1</b>	<b>CICSplex SM component of CICS TS 2.3</b>	<b>CICSplex SM component of CICS TS 2.2</b>
CICS TS 3.2	Yes	No	No	No
CICS TS 3.1	Yes	Yes	No	No
CICS TS 2.3	Yes	Yes	Yes	No
CICS TS 2.2	Yes	Yes	Yes	Yes
TXSeries 4.3.0.4	No	No	Yes	Yes
TXSeries 5.0	No	No	Yes	Yes

---

## Summary of changes

This book is based on the CICS Installation Guide for CICS Transaction Server for z/OS, Version 3 Release 1, GC34-6426-00. Changes from that edition are marked by vertical bars in the left margin.

This part lists briefly the changes that have been made for the following recent releases:

---

### Changes for CICS Transaction Server for z/OS, Version 3 Release 2

For information about changes that have been made in CICS Transaction Server for z/OS, Version 3 Release 2, please refer to *What's New* in the information center, or the following publications:

- *CICS Transaction Server for z/OS Release Guide*
- *CICS Transaction Server for z/OS Migration from CICS TS Version 3.1*
- *CICS Transaction Server for z/OS Migration from CICS TS Version 2.3*
- *CICS Transaction Server for z/OS Migration from CICS TS Version 2.2*
- *CICS Transaction Server for z/OS Migration from CICS TS Version 1.3*

---

### Changes for CICS Transaction Server for z/OS, Version 3 Release 1

The more significant changes for this edition are:

- Technical changes :
  - A new topic, Chapter 28, “Enabling Unicode data conversion by z/OS,” on page 205
  - There are changes to Chapter 6, “Installing the CICS Information Center,” on page 73
  - In CICSplex SM, support for the CICS NT remote MAS is removed, and the corresponding topic in the section about CICSplex SM verification is deleted.
  - References to z/OS UNIX directories now show the optional */pathprefix*, see, for example, “Specify the CICS TS z/OS UNIX directories and data sets” on page 34
- Structural changes :
  - Chapter 5, “Installing CICS TS using DFHISTAR,” on page 25 is a collection of topics which used to appear in the *CICS Transaction Server for z/OS Program Directory*
  - “Setting up a CICSplex SM Web User Interface server” on page 284 is a topic which used to appear in the the *CICSplex System Manager Web User Interface Guide*

---

### Changes for CICS Transaction Server for z/OS Version 2 Release 3

The more significant changes for this edition are:

- Technical changes :
  - In CICSplex SM, support for the CICS OS/2 remote MAS is removed, and the corresponding topic in the section about CICSplex SM verification is deleted.
  - New or revised topics include:
    - “Authorizing the hlq.SDFJAUTH library” on page 238

- “Populating the CICSplex SM data repository” on page 259
- “CMAS-related CICS SIT parameters” on page 263
- “Preparing the MAS for history recording” on page 282
- Structural changes :
  - Chapter 6, “Installing the CICS Information Center,” on page 73 has been moved from the back of the book. It is now in Chapter 1, “Planning for installation,” on page 3.

---

## Changes for CICS Transaction Server for z/OS Version 2 Release 2

The more significant changes for this edition are:

- Technical changes :
  - In support of the new CICSplex SM remote MAS agent for Windows, two chapters have been added:
    - Setting up a CICSplex SM remote MAS agent for Windows
    - Installation verification procedure 6 (IVP6)
  - “CICSplex SM system parameters” on page 305 has been updated with several new security-related parameters.
- “Setting up a CICSplex SM address space (CMAS)” on page 252 has been expanded with the addition of the new sections “Controlling tasks in a CMAS” on page 266 and “Controlling CICS storage in a CMAS” on page 269.
- Structural changes :
  - A new chapter “Enabling REXX for CICS” on page 245 contains information about REXX that previously appeared in the *CICS Transaction Server for z/OS Program Directory*.
  - The former Appendix A, dealing with “Disk space needed for CICS”, has been removed. This information is available from the *CICS Transaction Server for z/OS Program Directory*. As a consequence, the former Appendix B is now Appendix A.

---

## Changes for CICS Transaction Server for z/OS, Version 2 Release 1

The more significant changes for this edition are:

- This book has a new section, Chapter 1, “Planning for installation,” on page 3, which replaces the *Planning for Installation* manual of earlier releases.
- Information has been added to Chapter 8, “Authorizing CICS regions to access MVS resources,” on page 81 describing “Authorizations for users of IXCMIAPU” on page 84.
- In Chapter 27, “Defining the logger environment for CICS journaling,” on page 169, suggested values for HIGHOFFLOAD and LOWOFFLOAD have been revised, and new sections “Requirements planning and checklist” on page 169 and “Analyzing SMF Type 88 records” on page 201 have been added.
- “Defining DL/I support” on page 230 is significantly changed.
- “Adding CICS support for programming languages” on page 232 is new.
- “Verifying your Java components installation” on page 236, replaces the previous “Java support” chapter.
- “Enabling TCP/IP in a CICS region” on page 243 is new.
- CICSplex SM for this release of CICS Transaction Server does not support CICS systems running under VSE/ESA. The chapters: “Setting up a CICS/VSE remote managed application system (MAS)” and “Installation verification procedure 3 (IVP3)”, (of the previous release) are removed.

For most items in each of the lists above, there is a reference to the part of the book where there is more detail.

---

## Changes for CICS Transaction Server for OS/390 Version 1 Release 3

- Post-installation, addition to chapter on authorizing CICS regions
- VTAM<sup>®</sup> definitions required for CICS: PERSIST=MULTI
- Defining an MVS console
- Java support

### CICSplex SM installation and setup

To support the inclusion of CICSplex SM as an element of CICS Transaction Server for OS/390<sup>®</sup>, Version 1 Release 3, and the consequent revised installation processes, information about how to install and verify the CICSplex SM element has been added to this book. This information was previously available in the *CICSplex SM Setup* book.



---

## **Part 1. Planning for Installation**

This part discusses planning considerations before installation.





---

## Chapter 1. Planning for installation

This section describes the areas you need to consider when planning your installation.

It contains the following topics:

- “Overview of CICS TS”
- “Packaging”
- “CICS TS elements and features” on page 4
- “What you receive with CICS TS” on page 4
- “Documentation” on page 6
- “The CICS Clients and The CICS Transaction Gateway” on page 6

---

### Introduction

Generally, large online CICS applications run on an MVS operating system together with a collection of other supporting software products, some provided by IBM and others by independent software vendors (ISVs). Functional additions include distributed CICS software on alternative platforms, enabling you to distribute transaction processing, with CICS on the MVS host acting as a large database server. IBM recognizes that customers traditionally run these products at various release levels—a piecemeal or mix-and-match approach.

CICS TS, together with other z/OS Software Servers, is designed to make it easier to install and operate the mix of software you need to run your business.

---

### Overview of CICS TS

With CICS Transaction Server for z/OS (CICS TS), IBM continues to integrate CICS with a set of other supporting software, offering you a single product in place of several products. You order an entire set of software, integrated into one licensed product, instead of having to order new levels of some products but not others.

CICS TS is a member of the family of MVS-based software servers, and is separately orderable as a single part number.

Within CICS TS, the levels of all products reflect the level of the CICS TS product itself. Even the word “product” has new meaning; for this reason the products that make up the base of CICS TS are called **elements**. CICS TS marks a significant change from the former piecemeal approach to the way you order and install CICS and its related software.

Getting all the elements in CICS TS installed and running is the subject of this book. Because the elements and features of CICS TS are integrated into a single package with compatible service levels, it is expected that you will migrate all elements and features of CICS TS at the same time.

---

### Packaging

The packaging of CICS TS into a number of elements is similar to the packaging of z/OS. For its operating system environment, CICS TS requires . As you prepare to install CICS TS, keep in mind that you gain the benefits of a comprehensively tested environment when you install CICS TS.

In summary, the concept of CICS TS is to integrate a range of transaction server functions into a single product that delivers the function previously provided by a number of individual IBM software products. CICS TS consists of several base elements. The intent is that IBM ships, and you run, all elements at the single release level that IBM has subjected to comprehensive system testing.

Migrating from one release of CICS TS to the next is relatively simple, and the transition from your current systems to CICS TS should also be straightforward.

---

## CICS TS elements and features

CICS TS consists of base elements that deliver essential transaction server functions. When you order CICS TS, you receive all the base elements, described in “What you receive with CICS TS.”

Two methods of installing CICS TS come free with your license:

1. A system replace method called ServerPac
2. The Custom-Built Product Delivery Option (CBPDO).

## Exclusive and non-exclusive elements and features

Some elements and features contain new function that is available only within CICS TS. This book labels such an element or feature **exclusive**: new function is exclusive to CICS TS. If an element or feature is **exclusive**, you receive new function only through CICS TS. That is, while prior levels continue to be available, future functional enhancements occur only within CICS TS.

Other elements, however, exist both within CICS TS and also as separately orderable products. These are **non-exclusive**.

IBM's direction is to make functional enhancements only within CICS TS.

---

## What you receive with CICS TS

Because the elements of CICS TS are integrated into a single package, you are expected to install the entire product.

You can install CICS TS using one of several IBM packages. Two of these packages are available at no additional charge when you license CICS TS: (1) ServerPac, the system replace deliverable, or (2) CBPDO.

There is no stand-alone product tape for CICS TS, and there is no Custom Built Installation Process Offering (CBIPO).

Because of the overall ease of installing, **IBM recommends** that you choose ServerPac, if possible.

- **If you order the ServerPac offering**, you receive:
    - A series of tapes, each in IEBCOPY dump-by-dataset format (not a physical volume dump) containing a complete generated CICS TS system. This consists of distribution and target libraries, consolidated software inventory (CSI) and other SMP/E libraries already generated. CICS TS elements and their service are integrated into distribution and target libraries.
- IBM has IPLed the system and executed all installation verification programs (IVPs) prior to shipment.

- A CustomPac dialog, accessed through ISPF, that produces jobs that unload the tape to DASD. Through the dialog, you can name the data sets and place them in the catalogs you specify. The following accompanying documentation tells you how to use the dialog:
    - *ServerPac: Installing Your Order* (customized for each customer order)
    - *ServerPac: Using the Installation Dialog*, SA22-7815
  - All unintegrated service, available on a service tape.
  - Sample jobs to assist with the installation of CICS TS product and service.
- Through the dialog, you can:
- Name the data sets and place them on the volumes and in the catalogs you choose
  - Save configuration data for the next install, easing your move to the next release of CICS TS
  - Run tailored post-installation verification jobs.
- **If you order CBPDO**, you receive one logically stacked SMP/E RELFILE tape that contains all the base elements. Depending on your customer profile, you receive uninstalled service. You also receive:
    - Sample jobs to help you install CICS TS and service.
    - *Custom-Built Offerings Plan/Install*, SC23-0352; the CBPDO Memo to Users Extension; CICS TS *Program Directory*, GC33-1200, and program materials that describe your CBPDO order.

Figure 1 on page 6 illustrates the content differences between the ServerPac method and the CBPDO method of installing CICS TS.

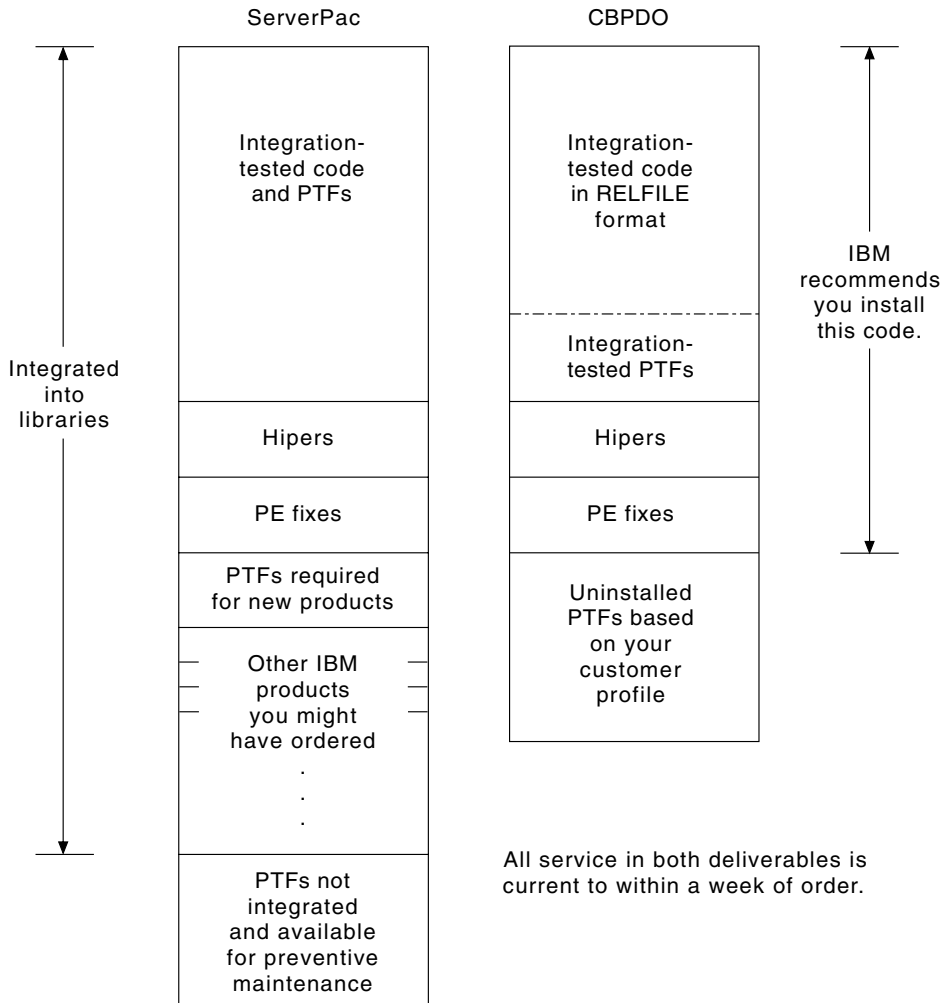


Figure 1. Contents of the ServerPac and the CBPDO Delivery

---

## Documentation

Whether you use ServerPac or CBPDO, you receive:

- The CICS TS Program Directory and other installation information for the elements, in hardcopy and softcopy.
- Essential hardcopy books to help you with installation.
- All the publications for CICS Transaction Server for z/OS in softcopy as part of the CICS Information Center. This is supplied on a CD-ROM. Installation instructions for the CICS Information Center can be found in Chapter 6, "Installing the CICS Information Center," on page 73.

---

## The CICS Clients and The CICS Transaction Gateway

CICS Transaction Server for z/OS does not include the CICS Universal Clients or the CICS Transaction Gateway. To use the CICS Universal Client V5 or the CICS Transaction Gateway V5, licenses for these products are required.

---

## Chapter 2. Requirements for CICS TS

A major part of your planning effort involves looking at the software and hardware required for the system that you are installing.

---

### Hardware requirements for installing CICS TS

Whether you choose the CBPDO method or the ServerPac method, the hardware requirement is the same.

Hardware requirements are more fully described in Hardware requirements in the *CICS Transaction Server for z/OS Release Guide*

You need a hardware configuration that runs the required levels of MVS, provided the configuration has a terminal and a tape device capable of reading one of the following types of tape on which CICS TS is supplied:

- 6250 bpi 9-track reels
- 3480 cartridges
- 4MM DAT cartridges

---

### Software requirements for installing CICS TS

The system software requirements for installing CICS TS using the ServerPac method or the CBPDO method are the same except for the addition of SMP/E for CBPDO.

Software requirements are more fully described in the *CICS Transaction Server for z/OS Release Guide*.

The products shown in Table 2 must be installed on the system you use to install both the ServerPac and the CBPDO.

*Table 2. Program products required on the installing system for ServerPac and CBPDO*

Program Product	Minimum Level
z/OS (5694-A01)	<p>z/OS, Version 1 Release 7</p> <p>z/OS includes the following elements that are required for installing CICS TS:</p> <ul style="list-style-type: none"><li>• Interactive System Productivity Facility (ISPF)</li><li>• Time Sharing Option/Extended (TSO/E)</li><li>• DFSMS/MVS</li><li>• Language Environment®</li><li>• Two components of eNetwork Communications Server: SNA and IP (previously VTAM and TCP/IP).</li><li>• UNIX system services (see “z/OS UNIX file system and PDSE requirements” on page 8 for more information).</li></ul> <p>If you are installing CICS TS using the CBPDO method, you also need:</p> <ul style="list-style-type: none"><li>• System Modification Program/Extended (SMP/E)</li><li>• High Level Assembler/MVS &amp; VM &amp; VSE.</li></ul>

**Service note:** See the *Program Directory for CICS Transaction Server for z/OS* for information about the service which should be applied on z/OS to ensure that CICS TS installs correctly.

---

## z/OS UNIX file system and PDSE requirements

Some components of CICS TS are installed in PDSE data sets and z/OS UNIX files. CICS TS for z/OS, Version 3.2 requires PDSE data set support for installation to complete successfully. DFSMS/MVS, which is supplied as an element of z/OS, provides this support for PDSE data sets. The components concerned are part of the CICS support for Java programs and for IIOB inbound to Java applications. For FMID JCI650D, which contains the z/OS UNIX-dependent code, the OMVS address space must be active in full-function mode, otherwise the install of this function fails.

In the set of installation jobs, there are some new initial jobs to create the z/OS UNIX files and the directories shown in Figure 2. You run these jobs before any of the normal DFHINST $n$  jobs. Note that the user ID running these jobs requires superuser authority.

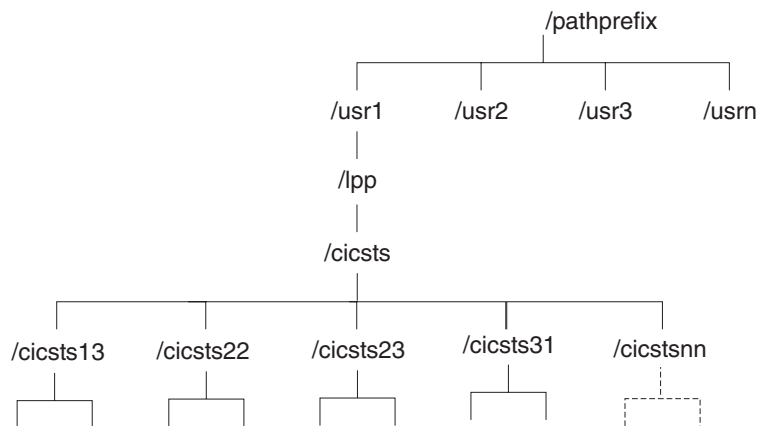


Figure 2. The z/OS UNIX directory structure for CICS java and IIOB files

**Note:**

1. If you normally maintain additional SMP/E target zone libraries for the purpose of applying service, you can also create additional directories at the /cicsts32 level to create the z/OS UNIX equivalent. See the DFHIHFSA job for more information.
2. The pathprefix in Figure 2 is optional.

---

## DASD storage requirements for CICS TS

The amount of storage required for all the **target** and **distribution** data sets is detailed in the space table in the *CICS Transaction Server for z/OS: Program Directory*.

## Chapter 3. Planning for CICSplex SM

*CICSplex System Manager Concepts and Planning* provides an introduction to CICSplex SM. If you are not already familiar with CICSplex SM, you should read it before you read this chapter.

### CICSplex SM setup checklist and worksheets

This topic contains the following aids to your installation and setup procedures:

#### Checklist

To use as a guide to your progress as you set up or revise the configuration of your CICSplex SM components. The checklist is for use with a CICS Transaction Server for z/OS system and all of the components you can install on it.

Some of the items on the MVS checklist need be performed only once for your CICSplex SM environment, while others must be performed once for each component. See the 'Where to get information' column for a reference to information about how to perform each task.

The order of items in the checklist is a suggested order for performing the installation and setup steps. However, you may find that, particularly if you are modifying your CICSplex SM environment, a different order is more practical.

#### Worksheets

To use as a record of the names and locations of components and data sets. The worksheets can be copied as you need.

The worksheets contain, in some cases, more than one line for a type of CICSplex SM component. You may have fewer or more than shown of that type of component.

A worksheet is provided for each of the following CICSplex SM system components:

- The CICSplex SM system
- A CMAS
- A MAS
- A WUI server

### MVS installation and setup checklist

Component			What you need to do	Values to note	Where to get information
<b>Note:</b> Components are indicated as follows: <b>CM</b> =CMAS, <b>M</b> =MAS, <b>W</b> =WUI server					
CM	M	W	Make note of SYS1.PARMLIB(IEASYSxx) values for this MVS system	APF= CMD= LNK= LNKAUTH= LPA= MAXCAD= MAXUSER= NSYSLX= PROG= RSVNONR= RSVSTRT= SMF= SYSNAME=	"Noting IEASYSxx values for CICSplex SM" on page 100
CM			Update number of common data spaces in IEASYSxx	NSYSLX value	"Updating IEASYSxx (CMAS)" on page 101
CM			Update number of linkage indexes in IEASYSxx	MAXCAD value	"Updating IEASYSxx (CMAS)" on page 101

Component			What you need to do	Values to note	Where to get information
CM	M	W	Update IEAAPFxx or PROGxx to authorize SEYUAUTH	IEAAPFxx or PROGxx member Library name	Chapter 7, "Authorizing the CICS and CICSplex SM libraries," on page 79
	M	W	Update IEAAPFxx or PROGxx to authorize SEYULPA Optional library. Can be populated below.	IEAAPFxx or PROGxx member Library name	Chapter 7, "Authorizing the CICS and CICSplex SM libraries," on page 79
CM			Verify SEYULINK is authorized	LNKAUTH= value Library name	Chapter 7, "Authorizing the CICS and CICSplex SM libraries," on page 79
CM			Update linklist with SEYULINK	LNKLSTxx member Library name	"CICS- and CICSplex SM-supplied modules required in the MVS linklist" on page 93
	M	W	Update LPA list with lindex.SEYULPA Optional library. Can be populated below.	LPALSTxx member Library name	"Installing CICSplex SM modules into the LPA" on page 141
CM	M	W	Use your ESM to protect CICSplex SM libraries	As required by your ESM	See the <i>CICS RACF Security Guide</i>
CM			Define security for the CMAS startup procedures	Procedure names	<i>CICS RACF Security Guide</i>
CM			Create VTAM application definition for each CMAS	SYS1.VTAMLST major node member Application name(s)	"Step 1: Creating a VTAM application definition for a CMAS" on page 125
CM			Define cross-domain resources for each CMAS	SYS1.VTAMLST members	"Step 2: Defining cross-domain resources for a CMAS" on page 125
CM	M	W	Edit DFHISTAR for post-installation members	Edited member	"CICSplex SM post-installation members" on page 217
CM	M	W	Run edited DFHISTAR member to generate POST install members.	sysproc.XEYUINST output library name	"CICSplex SM post-installation members" on page 217
	M	W	(Optional.) Install LPA modules	Installed usermod name	"Installing CICSplex SM modules into the LPA" on page 141
CM			Create CICSplex SM data repository	dsindex.EYUDREP.cmasname	"Creating the CICSplex SM data repository" on page 257
CM			Create CICSplex SM system parameter member for each CMAS	Modified EYUCMS0P parameter member(s)	"Preparing to start a CMAS" on page 254
	M		Edit CICSplex SM system parameter member for each MAS	Modified EYULMS0P parameter member(s)	"Preparing a MVS MAS" on page 276
		W	Edit CICSplex SM system parameter member for each WUI	Modified EYUWUI0P parameter member(s)	topics/eyuadih.dita#eyuadih
CM			Edit CICS SIT parameters for each CMAS	Modified parameter member(s)	"CMAS-related CICS SIT parameters" on page 263
	M		Edit CICS SIT parameters for each MAS	Modified parameter member(s)	"MVS MAS-related CICS SIT parameters" on page 280
		W	Edit CICS SIT parameters for each WUI	Modified parameter member(s)	topics/eyuadip.dita#eyuadip



Component		What you need to do	Values to note	Where to get information
CM		Create the CICS data sets and the data repository for each CMAS	Modified EYUCMSDS member	"Preparing to start a CMAS" on page 254
CM		Install CMAS startup procedure (EYUCMASP sample procedure)	Installed procedure member	"Preparing to start a CMAS" on page 254
	CM	Create CMAS-to-CMAS links using the WUI or the EYU9XDBT utility	CMAS names Target APPLID Target CICS SYSID	<i>CICSplex System Manager Administration</i>
CM		Create CICSplex definition using the WUI or the EYU9XDBT utility	CICSplex name	<i>CICSplex System Manager Administration</i>
	M	Create all MAS definitions using the WUI or the EYU9XDBT utility	MAS name(s)	<i>CICSplex System Manager Administration</i>
	M	Start the MAS	Message EYUXL0007I LMAS Phase II initialization complete	"Preparing a MVS MAS" on page 276
	M	Shut down the MASs using the WUI CICS regions view - terminates CICS	Message EYUXL0016I MAS shutdown complete	"Stopping management of a CICS system" on page 283

## System worksheet

System: \_\_\_\_\_

Subsystem ID: \_\_\_\_\_

VTAM Applid: \_\_\_\_\_

	Name:	VTAM Applid:	CICS-SYSID:
CMAS:	_____	_____	_____
WUI server:	_____	_____	_____
MAS:	_____	_____	_____
MAS:	_____	_____	_____
MAS:	_____	_____	_____
CMAS:	_____	_____	_____
WUI server:	_____	_____	_____
MAS:	_____	_____	_____
MAS:	_____	_____	_____
MAS:	_____	_____	_____
CMAS:	_____	_____	_____

	WUI server:	_____	_____	_____
	MAS:	_____	_____	_____
	MAS:	_____	_____	_____
	MAS:	_____	_____	_____

## CMAS worksheet

System: \_\_\_\_\_ CMAS name: \_\_\_\_\_  
 CICS Sysid: \_\_\_\_\_ VTAM Applid: \_\_\_\_\_

SYS1.PARMLIB(IEASYSxx) values:

APF= _____	LNK= _____	LNKAUTH= _____
MAXCAD= _____	NSYLSX= _____	PROG= _____

Dsn added to member IEAAPFxx or PROGxx: when \_\_\_\_\_ .SEYUAUTH  
 LNKAUTH=APFTAB: (when LNKAUTH=LNKLST, no dsn here) \_\_\_\_\_ .SEYULINK

Dsn added to member LNKLSTxx: \_\_\_\_\_ .SEYULINK

VTAM mode table node name: \_\_\_\_\_

SYS1.VTAMLST start list (ATCSTRxx): \_\_\_\_\_

SYS1.VTAMLST configuration list (ATCCONxx): \_\_\_\_\_

SYS1.VTAMLST applications member: \_\_\_\_\_

SYS1.VTAMLST cross-domain member: \_\_\_\_\_

VTAM definitions; node names:

	Name	VTAM Applid	CICS Sysid
MAS:	_____	_____	_____
MAS:	_____	_____	_____
MAS:	_____	_____	_____

Installation materials library: \_\_\_\_\_ .SDFHINST

Modified DFHISTAR (post-installation) member: \_\_\_\_\_

DFHINST exec output library: \_\_\_\_\_ .XDFHINST

CICSplex SM data repository dsn: \_\_\_\_\_

CICS resource definition tables output dsn: \_\_\_\_\_

CICS CSD dsn: \_\_\_\_\_

Modified EYUCMSOP member: \_\_\_\_\_

CICS SIT parameters member: \_\_\_\_\_

| Modified EYUCMSDS member: \_\_\_\_\_

CMAS startup procedure member: \_\_\_\_\_

Links to other CMASs:

CMAS name:	VTAM Applid:	CICS Sysid:	Protocol:
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____

Links to MASs:

MAS name:	VTAM Applid:	CICS Sysid:	Protocol:
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____
_____	_____	_____	_____

## MAS worksheet

MVS/ESA System: \_\_\_\_\_  
CICSplex Name: \_\_\_\_\_  
| CPSM Name: \_\_\_\_\_ MAS CPSM Name: \_\_\_\_\_  
| CICS Sysid: \_\_\_\_\_ MAS CICS Sysid: \_\_\_\_\_  
| VTAM Applid: \_\_\_\_\_ MAS VTAM Applid: \_\_\_\_\_  
MAS Type \_\_\_\_\_ AOR, TOR

Dsn added to member IEAAPFxx or PROGxx: when \_\_\_\_\_ .SEYUAUTH  
LNKAUTH=APFTAB: (when LNKAUTH=LNKLST, no dsn here)  
Dsn added to member LPALSTxx: \_\_\_\_\_ .SEYULPA  
SYS1.VTAMLST start list (ATCSTRxx): \_\_\_\_\_  
SYS1.VTAMLST configuration list (ATCCONxx): \_\_\_\_\_  
SYS1.VTAMLST applications member: \_\_\_\_\_  
Installation materials library: \_\_\_\_\_ .SDFHINST  
Modified DFHISTAR (post-installation) member: \_\_\_\_\_  
DFHINST exec output library: \_\_\_\_\_ .XDFHINST  
LPA module (usermod) name: \_\_\_\_\_  
CICS resource definition tables output dsn: \_\_\_\_\_  
  
CICS CSD dsn: \_\_\_\_\_  
Modified EYULMS0P dsn (member): \_\_\_\_\_  
CICS SIT parameters dsn (member): \_\_\_\_\_  
Link from CMAS: \_\_\_\_\_

CMAS name:	VTAM Applid:	CICS Sysid:	Protocol:
_____	_____	_____	_____

## WUI server worksheet

MVS/ESA System: \_\_\_\_\_  
CICSplex Name: \_\_\_\_\_  
| CPSM Name: \_\_\_\_\_ MAS CPSM Name: \_\_\_\_\_  
| CICS Sysid: \_\_\_\_\_ MAS CICS Sysid: \_\_\_\_\_  
| VTAM Applid: \_\_\_\_\_ MAS VTAM Applid: \_\_\_\_\_  
|

| Dsn added to member IEAAPFxx or PROGxx: when \_\_\_\_\_ .SEYUAUTH  
| LNKAUTH=APFTAB: (when LNKAUTH=LNKLST, no dsn here)  
| Dsn added to member LPALSTxx: \_\_\_\_\_ .SEYULPA  
| SYS1.VTAMLST start list (ATCSTRxx): \_\_\_\_\_

| SYS1.VTAMLST configuration list (ATCCONxx): \_\_\_\_\_  
 | SYS1.VTAMLST applications member: \_\_\_\_\_  
 | Installation materials library: \_\_\_\_\_SDFHINST  
 | Modified DFHISTAR (post-installation) member: \_\_\_\_\_  
 | DFHINST exec output library: \_\_\_\_\_XDFHINST  
 | LPA module (usermod) name: \_\_\_\_\_  
 | CICS resource definition tables output dsn: \_\_\_\_\_  
 |  
 | CICS CSD dsn: \_\_\_\_\_  
 | Modified EYUWUI0P dsn (member): \_\_\_\_\_  
 | CICS SIT parameters dsn (member): \_\_\_\_\_  
 | Link from CMAS: \_\_\_\_\_  
 |  
 |  
 | \_\_\_\_\_ CMAS name: \_\_\_\_\_ VTAM Applid: \_\_\_\_\_ CICS Sysid: \_\_\_\_\_ Protocol: \_\_\_\_\_

## IVP planning worksheet

Table 3.

Component	DFHISTAR parameters	Your value	DFHISTAR defaults
Load library high level qualifier for CICS	TINDEX.CICS. XTRAQUAL		CICSTS32.CICS
Load library high level qualifier for CICSplex SM	TINDEX.CPSM. XTRAQUAL		CICSTS32.CPSM
Attributes of the CICS TS system data sets	DSINFO		CICSTS32 CICS32 3390
Data set name of the SCEECICS library	SCEECICS		SYS1.SCEECICS
Data set name of the SCEERUN library	SCEERUN		SYS1.SCEERUN
Data set name of the SCEERUN2 library	SCEERUN2		SYS1.SCEERUN2
Data set name of the SCEESAMP library	SCEESAMP		SYS1.SCEESAMP
CMAS name	CMASNAME		CMAS01
CMAS sysid	CMSSYSID		CM01
CMAS applid	CMASNAME		CMAS01
CICSplex name	CSYSPLEX		CSYPLX01
MAS name	CSYSNAME		CSYS01
MAS sysid	CSYSYSID		CS01
Web User Interface CICSplex name	WUIPLEX		WUIPCM01
Web User Interface applid	WUINAME		WUINCM01
Web User Interface sysid	WUISYSID		WU01

Table 3. (continued)

Component	DFHISTAR parameters	Your value	DFHISTAR defaults
TCP/IP host name	TCPIPHST		XXXXXXXX. XXXXXXXX. XXXXXXXX. XXXXXXXX
TCP/IP port for Web User Interface	TCPIPPRT		12345
Timezone	TIMEZONE		B

---

## Planning for CICSplex SM migration

CICS TS for z/OS, Version 3.2 CICSplex SM can manage CICS regions running under the following releases of CICS:

- CICS Transaction Server for z/OS, Version 3 Release 1
- CICS Transaction Server for z/OS, Version 2 Release 3
- CICS Transaction Server for z/OS, Version 2 Release 2

For detailed information about migrating to the CICS Transaction Server for z/OS, Version 3 Release 2 level of CICSplex SM, see *CICS Transaction Server for z/OS Migration from CICS TS Version 3.1* or equivalent for your level of CICSplex SM.



---

## Part 2. Installation

This part discusses installation.





---

## Chapter 4. Installing CICS TS

IBM offers the following methods for installing CICS TS:

- ServerPac
- CBPDO, using the SMP/E RECEIVE, APPLY, and ACCEPT commands
- CBPDO, using the DFHISTAR process
- IBM customized packages.

The first three of these are entitled offerings, and are the subject of this book, but IBM customized packages are fee-based and are not discussed except for the following summary information. Depending on the country in which you order, you can purchase one of the following customized packages:

- A SystemPac®, which tailors CICS TS to your environment, such as DASD layout and naming conventions, based on information provided to IBM.
- Other fee-based services and customized offerings.

Both the ServerPac and CBPDO methods of delivering CICS TS come with a set of documentation that you use when you install the product. To help you plan ahead, this section gives you a preview of some of this information, such as the information contained in the *Program Directory*, which is shipped regardless of the delivery method you choose.

**Note:** The *Program Directory* is available in hardcopy only with the product—it cannot be ordered separately. A softcopy version is available on the CICS TS CD-ROM product kit, and also on the Transaction Processing and Data Collection Kit, SK2T-0730. The *Program Directory* is also available on the CBPDO and ServerPac tapes.

This chapter also helps you with other planning steps you must take:

- Ensuring you have the required hardware and software to install and run CICS TS; see Chapter 2, “Requirements for CICS TS,” on page 7.
- Ensuring you have enough DASD storage space for CICS TS; see “DASD storage requirements for CICS TS” on page 8.
- Outlining the install steps for CBPDO; see “Installing CBPDO”

---

### Installing CBPDO

You install all the elements from the CICS TS CBPDO using a single installation process. The CBPDO *Memo to Users Extension* contains information about the CBPDO you ordered, and the features and service it includes. It also contains CBPDO installation information.

The first planning task is to read the *Memo to Users Extension* thoroughly before starting any of the install tasks. If you are a new user of CBPDO, you should also read the IBM publication, *MVS Custom-Built Offerings Planning and Installation*, SC23-0352.

The *CICS Transaction Server for z/OS: Program Directory* gives a sample IEBCOPY job that you can customize to copy RELFILE(2) from the CICS TS CBPDO tape. Modify the LABEL=3 parameter to reference the file number of RELFILE(2) as supplied on the CBPDO tape. When you have copied RELFILE(2) to

DASD, you generate a single set of install jobs using the CICS TS job generator, DFHISTAR. This generates the following set of customized install jobs, based on the parameters you specify to DFHISTAR:

- DFHIHFS0, DFHIHFS1, DFHIHFSA, and DFHISMKD, the z/OS UNIX-related jobs
- DFHINST1 through DFHINST6

## Using the SMP/E RECEIVE, APPLY, and ACCEPT commands

The *CICS Transaction Server for z/OS: Program Directory* describes this process and tells you when to return to the *CICS Transaction Server for z/OS Installation Guide* to continue.

## Using the DFHISTAR process

“Edit the DFHISTAR Job” on page 25 describes all the parameters that you can specify to customize the install jobs, and explains the jobs that are generated.

One of the most significant parameters you are asked to specify is the high-level qualifier for the data sets into which the jobs install the product. CICS TS comprises a number of elements that are installed in a single process by the DFHINST $n$  jobs. To ensure the element libraries are easily identified, DFHISTAR adds an element qualifier to the data set names. Using the default high-level qualifier CICSTS32, the names generated by DFHISTAR take the following form:

**CICS**   CICSTS32.CICS.*ddname*

**CICSplex SM**  
          CICSTS32.CPSM.*ddname*

**Application Migration Aid**  
          CICSTS32.AMA.*ddname*

**REXX for CICS**  
          CICSTS32.REXX.*ddname*

To enable you to customize the z/OS UNIX-related jobs, the following parameters are provided in the DFHISTAR job:

### HFS0DSN

The data set name of the file system to be mounted at directory */pathprefix***usr/lpp/cicsts**. Pathprefix is variable, and optional. The other parts of this directory name structure are fixed. The default data set name is OMVS.USR.LPP.CICSTS.

This parameter is used by job DFHIHFS0.

### HFS1DSN

The data set name of the file system to be mounted at directory */pathprefix***usr/lpp/cicsts/ussdir**, where *ussdir* is a variable you specify in the **ussdir** parameter in DFHISTAR. If you omit the **ussdir** parameter it defaults to the value of the TINDEXT parameter in lowercase, which in turn defaults to CICSTS32, so if both defaults are taken, the full directory path resolves to */pathprefix***usr/lpp/cicsts/cicsts32**.

The default data set name is OMVS.USR.LPP.CICSTS.CICSTS32.

This parameter is used by job DFHIHFS1.

### HFSADSN

The data set name of the file system to be mounted at directory

`/pathprefix/usr/lpp/cicsts/ussdira`, where `ussdira` is a variable you also specify in DFHISTAR. If you omit the `ussdira` parameter it defaults to the value (in lower case) of AINDEX, which in turn defaults to CICSTS32.A. The default data set name is OMVS.USR.LPP.CICSTS.CICSTS32.A.

This parameter is used by job DFHIHFSA.

## Running the install jobs

Run the install jobs as follows:

1. Run the z/OS UNIX-related jobs to create the z/OS UNIX directories down to the `/cicsts32` level:

### DFHIHFS0 (required once only)

This job creates the file system specified on the HFS0DSN parameter and also the `/cicsts` directory at `/pathprefix/usr/lpp`.

This job is required once only, the first time you install CICS TS, and can be skipped in subsequent releases.

**Note:** When you install a new release of , directories down to the `/pathprefix/usr/lpp` level are replaced, effectively losing the `/cicsts` and lower directories. Issue the make directory command (`mkdir /pathprefix/usr/lpp/cicsts`) to recreate mount point `/pathprefix/usr/lpp/cicsts` for the file system defined in DFHIHFS0 (OMVS.USR.LPP.CICSTS). The mount command should already be in the PARMLIB member BPXPRMxx, copied from the DFHBPXP0 member of SDFHINST.

### DFHIHFS1 (required)

This job creates, at `/pathprefix/usr/lpp/cicsts`, the file system specified on the HFS1DSN parameter and also the directory specified on the parameter (default name `/cicsts32`).

### DFHIHFSA (optional)

This job creates, at `/usr/lpp/cicsts`, the alternate file system specified on the HFSADSN parameter and also the directory specified on the `ussdira` parameter (default name `/cicsts32.a`).

2. Run DFHISMKD to create the directories and file system under `/pathprefix/usr/lpp/cicsts/cicsts32`, required for FMID JCI650D, to contain the Java and IIOB classes, samples, and so on, that have to reside in the z/OS UNIX file system.
3. Run the DFHINST1 through DFHINST4 jobs as described in the *Program Directory*
4. Run the CBPDO-supplied SMP/E RECEIVE job, RCVPDO, located in the CBPDO RIMLIB dataset (this replaces the DFHINST5 job described in the *Program Directory*).
5. Run the DFHINST6 job.

**Note:** This job must run in the same MVS image in which you ran the DFHISMKD job to create the z/OS UNIX directories. DFHINST6 uses the CICS TS z/OS UNIX directories and data sets, and these are accessible only in the MVS in which you created them. This note does NOT apply if you are set up for MVS sharing.

There is more detailed information about all these jobs in the CICS TS *CICS Transaction Server for z/OS Program Directory*, and also in the comments at the start of each job.

On completion of the installation jobs, you have all the elements installed. Note that there is no provision within the DFHISTAR job generator, or in the generated jobs, to exclude an element from the install process. When you run the jobs, SMP/E installs all the elements included on the CICS TS CBPDO tape.

By default, SMP/E installs CICS TS in new SMP/E global, target, and distribution zones. The DFHINST3 job creates a new CSI data set for each zone. If you want to vary this default SMP/E configuration, see the *CICS Transaction Server for z/OS Program Directory* for information about the parameters that control the SMP/E zones and the disposition of the CSI data sets.

---

## Installing ServerPac

A CICS TS ServerPac consists of a number of tapes, the exact number depending on whether other products are included with the CICS TS Serverpac, and also on the type of tape requested. For example, a ServerPac order can comprise the following:

- A tape that contains related-installation material (RIM) files.
- Three tapes that contain the CICS TS product, consisting of all the SMP/E CSI data sets, and the target and distribution libraries.
- A service tape.

If you already have printed copies of the ServerPac manuals that you need to install the ServerPac, use these to guide you through the installation process. If you don't have copies, download and print the manuals from the first ServerPac tape. There is a sample job, in member PRTDOC, on the RIM tape, that you can use to print the manuals.

## What you need to install the ServerPac

You need the following resources to install the CICS TS ServerPac:

- A tape drive for reading the tapes.
- A TSO session for running the CustomPac dialog.
- A copy of the *ServerPac: Using the Installation Dialog* manual, SA22-7815.
- The *ServerPac: Installing Your Order*, customized for each customer order
- The CustomPac dialog supplied with ServerPac.

### First-time user of the CustomPac dialog for ServerPac

If you are installing a ServerPac for the first time, start by installing the CustomPac dialog, as described in the *ServerPac: Using the Installation Dialog* manual in “Chapter 2. Installing and Starting the Dialogs”.

When you have installed the dialog, invoke the dialog as directed in the *ServerPac: Using the Installation Dialog* manual under the INVOCATION topic.

### Existing user of the CustomPac dialog for ServerPac

If you have installed a ServerPac version of CICS TS for a previous order, use the dialog already installed. Invoke the CustomPac dialog, either from the ISPF primary options menu, or by invoking the ServerPac CLIST.

As an existing user of the CustomPac dialog, you can begin at the step described in the *ServerPac: Using the Installation Dialog* manual, in “Chapter 2. Receive a New Order”.

## Summary of the ServerPac installation steps

When you invoke the CustomPac dialog, you are presented with the primary option menu, from which you can:

- Receive the order (option **R**)
- Install the order (option **I**).

**Note:** The primary menu also enables you to display information about orders.

The following tasks are described, with illustrations of the various panels, in the *ServerPac: Using the Installation Dialog* manual:

### Receive the order

This step is described in section “6.0 Receive an Order”.

The “Order Receive” panels enable you enter your CICS TS order information, and complete the job card information needed to generate the order-receive job. The final phase of this step presents you with the generated JCL in an edit session, from which you can submit the job to receive the installation material (RIM) files to your DASD.

### Install orders

This step is described in section “7.0 Installation Menu”.

The “Installation Menu” panels enable you to:

- Configure the order control information tables.
- Define values for the installation variables used in skeleton batch jobs.
- Define the names of the SMP/E zones into which you want to install CICS TS.
- Modify data set profiles and DASD allocation for the order.
- Define catalog data set names and the aliases associated with them.
- Define system-specific aliases (where applicable)
- Select and submit the installation jobs
- Save the installation work configuration for use with future CICS TS orders
- Update the order inventory status.

---

## The next step

After you have completed the basic installation process using one of these methods, the next steps are to plan for CICSplex SM, and to prepare the environment so that you can run the CICS installation verification procedures (IVPs) and the CICSplex SM IVPs.

These steps are discussed in:

- Chapter 3, “Planning for CICSplex SM,” on page 9
- Chapter 7, “Authorizing the CICS and CICSplex SM libraries,” on page 79
- Chapter 31, “Getting ready to run CICS,” on page 215



---

## Chapter 5. Installing CICS TS using DFHISTAR

IBM offers the DFHISTAR process for those who prefer to use it for installation.

This topic covers the following subjects:

- “Edit the DFHISTAR Job”
- “Create RACF Profiles for the CICS Transaction Server Data Sets” on page 53
- “Run the DFHISTAR Job” on page 53
- “Check that You Are Ready to Run the Installation Jobs” on page 55
- “Run the Installation Jobs” on page 56
- “Check the Output from the Installation Jobs” on page 61
- “What next?” on page 61
- “Activating CICS Transaction Server” on page 66
- “Checklist for the CICS Transaction Server for z/OS Installation” on page 67

---

### Edit the DFHISTAR Job

Edit the DFHISTAR job, to assign values to installation parameters for your environment. The DFHISTAR job is in the TDFHINST library when you copy RELFILE(2) from the distribution tape. You can either edit the DFHISTAR job directly, or copy the DFHISTAR job (to preserve the IBM-supplied values) and edit your copy.

Remember that the product is installed using **cataloged** data sets.

This topic describes the process of editing the parameters in the DFHISTAR job. It contains the following sub-topics in an order that corresponds to the order of parameters in the DFHISTAR job:

- “Specify the CICS Transaction Server temporary installation libraries” on page 31
- “Specify the JOB parameters for installation jobs” on page 31
- “Specify the scope of the installation” on page 32
- “Specify the type of JES to be used” on page 32
- “Specify the utilities to be used” on page 32
- “Specify the prefix of CICS Transaction Server jobs” on page 33
- “Specify the indexes of CICS Transaction Server data sets” on page 33
- “Specify the CICS TS z/OS UNIX directories and data sets” on page 34
- “Specify extra qualifiers” on page 35
- “Specify block sizes” on page 35
- “Specify the disk unit for work data sets” on page 35
- “Specify SMS option for DASD allocations” on page 35
- “Specify disk volumes” on page 36
- “Allocate space for CICS Transaction Server disk volumes” on page 39
- “Specify Attributes of the Temporary SMP/E Work Data Sets” on page 39
- “Specify Attributes of the Permanent SMP/E Data Sets” on page 40
- “Specify SMP/E Zone Attributes” on page 41
- “Specify the High-level Qualifiers for SMP/E Data Sets” on page 44
- “Specify the distribution tape device type” on page 44
- “Specify attributes of the CICS Transaction Server system data sets” on page 44
- “Specify Attributes of any additional target Libraries” on page 45

- “Specify the Data Set Name of the SISpload Library” on page 46
- “Specify the Data Set Name of the CSSLIB Library” on page 46
- “Specify the Data Set Name of the SCEECPP Library” on page 46
- “Specify the Data Set Name of the SCEELKED Library” on page 47
- “Specify the Data Set Name of the SCEELKEX Library” on page 47
- “Specify the Data Set Name of the SCEEOBJ Library” on page 47
- “Specify the Data Set Name of the SCEESAMP Library” on page 47
- “Specify the Data Set Name of the SCLBSID Library” on page 47
- “Specify the Data Set Name of the SCSQLOAD Library” on page 47
- “Specify the Data Set Name of the SCSQANLE Library” on page 48
- “Specify the Data Set Name of the SCSQCICS Library” on page 48
- “Specify the Data Set Name of the SCSQAUTH Library” on page 48
- “Specify the installation directory for Java support” on page 48
- “Specify the Data Set Name of the SDSNLOAD Library” on page 48
- “Specify the Data Set Name of the SEZARPCL and SEZACMTX libraries” on page 49
- “Specify the Data Set Name of the SCEECICS & SCEERUN libraries” on page 49
- “Specify the Data Set Name of the SCEERUN2 & SCEELIB libraries” on page 49
- “Specify the Data Set Name of the SCEEBND2 library” on page 50
- “Specify Log stream and Log stream Structure Attributes.” on page 50
- “Specify attributes specific to CICSplex SM” on page 50

To help you look up details about a particular parameter, Table 4 lists the DFHISTAR parameters, in alphabetical order, with their predefined values. (The parameters in the DFHISTAR job itself are listed in associated groups.) It also indicates whether a parameter is relevant to CICS, or CICSplex SM, or both.

The default values of the parameters are the same as the IBM-supplied values.

You can use the IBM-supplied values, define your own values, or let the CICS Transaction Server installation process determine default values for you.

**Note:** You can enter your values for parameters of the DFHISTAR job in lowercase; except for the z/OS UNIX-related parameters (USSDIR, USSDIRA, and JAVADIR), the values are translated into uppercase when you run the DFHISTAR job.

Table 4. Alphabetical list of parameters for the DFHISTAR job

Parameter	CICS	CICSplex SM	Supplied value	Refer to
ADDTVOL	X		CICSTS32 SYSALLDA	“Specify Attributes of any additional target Libraries” on page 45
AINDEX	X		CICSTS32.A	“Specify Attributes of any additional target Libraries” on page 45
ALINDEX	X		SYS1.CICSTS32.A	“Specify Attributes of any additional target Libraries” on page 45



Table 4. Alphabetical list of parameters for the DFHISTAR job (continued)

Parameter	CICS	CICSplex SM	Supplied value	Refer to
ASMPPTS	X		CICSTS32.A.SMPLTS	“Specify Attributes of any additional target Libraries” on page 45
ASMPMTS	X		CICSTS32.A.SMPMTS	“Specify Attributes of any additional target Libraries” on page 45
ASMPSCDS	X		CICSTS32.A.SMPSCDS	“Specify Attributes of any additional target Libraries” on page 45
ASMPSTS	X		CICSTS32.A.SMPSTS	“Specify Attributes of any additional target Libraries” on page 45
AZONE	X		AZONE	“Specify Attributes of any additional target Libraries” on page 45
AZONECSI	X		CICSTS32.A.AZONE	“Specify Attributes of any additional target Libraries” on page 45
AZONELOG	X		CICSTS32.A.AZONE.SMPLOG	“Specify Attributes of any additional target Libraries” on page 45
BLKFB80	X		0	“Specify block sizes” on page 35
BLKISPF				This parameter is now obsolete.
BLKU	X	X	32760	“Specify block sizes” on page 35
CMACVOL	X		CICSTS32	“Specify disk volumes” on page 36
CMASNAME		X	CMAS01	“Specify attributes specific to CICSplex SM” on page 50
CMSSYSID		X	CM01	“Specify attributes specific to CICSplex SM” on page 50
CSSLIB	X		SYS1.CSSLIB	“Specify the Data Set Name of the CSSLIB Library” on page 46
CSYSYSID		X	CS01	“Specify attributes specific to CICSplex SM” on page 50
CSYSNAME		X	CSYS01	“Specify attributes specific to CICSplex SM” on page 50
CSYSPLEX		X	CSYPLX01	“Specify attributes specific to CICSplex SM” on page 50
DEFVOL	X	X	CICSTS32 SYSALLDA	“Specify disk volumes” on page 36
DINDEX	X	X	CICSTS32	“Specify the indexes of CICS Transaction Server data sets” on page 33
DISTVOL	X	X	CICSTS32 SYSALLDA	“Specify disk volumes” on page 36
DOWNLOAD				This parameter is not used.
DSINFO	X	X	CICSTS32 CICSTS32 SYSALLDA .	“Specify attributes of the CICS Transaction Server system data sets” on page 44
DZONE	X		DZONE	“Specify SMP/E Zone Attributes” on page 41
DZONECSI	X		CICSTS32.DZONE NEW CICSTS32 SYSALLDA	“Specify SMP/E Zone Attributes” on page 41

Table 4. Alphabetical list of parameters for the DFHISTAR job (continued)

Parameter	CICS	CICSplex SM	Supplied value	Refer to
DZONELOG	X		CICSTS32.DZONE.SMPLOG NEW	“Specify SMP/E Zone Attributes” on page 41
GINDEX	X	X	CICSTS32	“Specify the indexes of CICS Transaction Server data sets” on page 33
GZONE	X		NEW CICSOPT	“Specify SMP/E Zone Attributes” on page 41
GZONECSI	X	X	CICSTS32.GZONE NEW CICSTS32 SYSALLDA	“Specify SMP/E Zone Attributes” on page 41
GZONELOG	X		CICSTS32.GZONE.SMPLOG NEW	“Specify SMP/E Zone Attributes” on page 41
HFS0DSN	X		OMVS.USR.LPP.CICSTS	“Specify the CICS TS z/OS UNIX directories and data sets” on page 34
HFS1DSN	X		OMVS.USR.LPP.CICSTS.CICSTS32	“Specify the CICS TS z/OS UNIX directories and data sets” on page 34
HFSADSN	X		OMVS.USR.LPP.CICSTS.CICSTS32..A	“Specify the CICS TS z/OS UNIX directories and data sets” on page 34
JAVADIR	X		java142/J1.4	“Specify the installation directory for Java support” on page 48
JES	X		JES2	“Specify the type of JES to be used” on page 32
JOB	X	X	(No valid value)	“Specify the JOB parameters for installation jobs” on page 31
LIB	X	X	CICSTS32.XDFHINST	“Specify the CICS Transaction Server temporary installation libraries” on page 31
LINDEX	X	X	SYS1.CICSTS32	“Specify the indexes of CICS Transaction Server data sets” on page 33
LOGGER-INFO	X		001 500 4096 64000 2048 MVSX XXXXXXXX XXXXXXXX	“Specify Log stream and Log stream Structure Attributes.” on page 50
NEWDREP		X		“Specify block sizes” on page 35
OLDDREP		X		“Specify attributes specific to CICSplex SM” on page 50
OPTVOL	X		CICSTS32 SYSALLDA	“Specify disk volumes” on page 36
PATHPREFIX	X		Null string	“Specify the CICS TS z/OS UNIX directories and data sets” on page 34
PREFIX	X		DFH	“Specify the prefix of CICS Transaction Server jobs” on page 33
SCEECICS	X	X	SYS1.SCEECICS	“Specify the Data Set Name of the SCEECICS & SCEERUN libraries” on page 49
SCEECPP	X		SYS1.SCEECPP	“Specify the Data Set Name of the SCEECPP Library” on page 46

Table 4. Alphabetical list of parameters for the DFHISTAR job (continued)

Parameter	CICS	CICSplex SM	Supplied value	Refer to
SCEELIB	X		SYS1.SCEELIB	“Specify the Data Set Name of the SCEECICS & SCEERUN libraries” on page 49
SCEEBND2	X		SYS1.SCEEBND2	“Specify the Data Set Name of the SCEECICS & SCEERUN libraries” on page 49
SCEERUN	X	X	SYS1.SCEERUN	“Specify the Data Set Name of the SCEECICS & SCEERUN libraries” on page 49
SCEERUN2	X	X	SYS1.SCEERUN2	“Specify the Data Set Name of the SCEECICS & SCEERUN libraries” on page 49
SCEELKED	X		SYS1.SCEELKED	“Specify the Data Set Name of the SCEELKED Library” on page 47
SCEELKEX	X		SYS1.SCEELKEX	“Specify the Data Set Name of the SCEELKEX Library” on page 47
SCEEOBJ	X		SYS1.SCEEOBJ	“Specify the Data Set Name of the SCEEOBJ Library” on page 47
SCEESAMP	X	X	SYS1.SCEESAMP	“Specify the Data Set Name of the SCEESAMP Library” on page 47
SCSFMOD0	X		SYS1.SCSFMOD0	“Specify the Data Set Name of the SCSFMOD0 and SIXMEXP libraries” on page 49
SCLBSID	X		SYS1.SCLBSID	“Specify the Data Set Name of the SCLBSID Library” on page 47
SCOPE	X	X	ALL	“Specify the scope of the installation” on page 32
SCSQLOAD	X		SYS1.SCSQLOAD	“Specify the Data Set Name of the SCSQLOAD Library” on page 47
SCSQANLE	X		SYS1.SCSQANLE	“Specify the Data Set Name of the SCSQANLE Library” on page 48
SCSQCICS	X		SYS1.SCSQCICS	“Specify the Data Set Name of the SCSQCICS Library” on page 48
SCSQAUTH	X		SYS1.SCSQAUTH	“Specify the Data Set Name of the SCSQAUTH Library” on page 48
SDSNLOAD	X		SYS1.SDSNLOAD	“Specify the Data Set Name of the SDSNLOAD Library” on page 48
SEZACMTX	X		SYS1.SEZACMTX	“Specify the Data Set Name of the SEZARPCL and SEZACMTX libraries” on page 49
SEZARPCL	X		SYS1.SEZARPCL	“Specify the Data Set Name of the SEZARPCL and SEZACMTX libraries” on page 49

Table 4. Alphabetical list of parameters for the DFHISTAR job (continued)

Parameter	CICS	CICSplex SM	Supplied value	Refer to
SISPLOAD	X		SYS1.SISPLOAD	“Specify the Data Set Name of the SISPLOAD Library” on page 46
SIXMEXP	X		SYS1.SIXMEXP	“Specify the Data Set Name of the SCSFMOD0 and SIXMEXP libraries” on page 49
SMPLTS	X		CICSTS32.SMPLTS	“Specify Attributes of the Permanent SMP/E Data Sets” on page 40
SMPMTS	X		CICSTS32.SMPMTS	“Specify Attributes of the Permanent SMP/E Data Sets” on page 40
SMPPTS	X		CICSTS32.SMPPTS	“Specify Attributes of the Permanent SMP/E Data Sets” on page 40
SMPSCDS	X		CICSTS32.SMPSCDS	“Specify Attributes of the Permanent SMP/E Data Sets” on page 40
SMPSTS	X		CICSTS32.SMPSTS	“Specify Attributes of the Permanent SMP/E Data Sets” on page 40
SMPVOL	X	X	CICSTS32 SYSALLDA	“Specify disk volumes” on page 36
SMPWORK	X	X	SYSALLDA	“Specify Attributes of the Temporary SMP/E Work Data Sets” on page 39
SMS	X		NO	“Specify SMS option for DASD allocations” on page 35
TAPEUNIT	X		3480	“Specify the distribution tape device type” on page 44
TARGVOL	X	X	CICSTS32 SYSALLDA	“Specify disk volumes” on page 36
TCPIPHST		X	XXXXXXXX.XXXXXXXXXX.XXXXXX XX.XXXXXXXXXX	“Specify attributes specific to CICSplex SM” on page 50
TCPIPPRT		X	12345	“Specify attributes specific to CICSplex SM” on page 50
TEMPLIB	X	X	CICSTS32.TDFHINST	“Specify the CICS Transaction Server temporary installation libraries” on page 31
TIMEZONE		X	B	“Specify attributes specific to CICSplex SM” on page 50
TINDEX	X	X	CICSTS32	“Specify the indexes of CICS Transaction Server data sets” on page 33
TZONE	X	X	TZONE	“Specify SMP/E Zone Attributes” on page 41
TZONECSI	X		CICSTS32.TZONE NEW CICSTS32 SYSALLDA	“Specify SMP/E Zone Attributes” on page 41
TZONELOG	X		CICSTS32.TZONE.SMPLOG NEW	“Specify SMP/E Zone Attributes” on page 41
USSDIR	X		.	“Specify the CICS TS z/OS UNIX directories and data sets” on page 34

Table 4. Alphabetical list of parameters for the DFHISTAR job (continued)

Parameter	CICS	CICSplex SM	Supplied value	Refer to
USSDIRA	X		.	“Specify Attributes of any additional target Libraries” on page 45
UTILITIES	X	X	ASMA90 IEWL GIMSMP IEBCOPY	“Specify the utilities to be used” on page 32
WORKUNIT	X	X	SYSALLDA	“Specify the disk unit for work data sets” on page 35
WUI		X	YES	“Specify attributes specific to CICSplex SM” on page 50
WUINAME		X	WUINCM01	“Specify attributes specific to CICSplex SM” on page 50
WUIPLEX		X	WUIPCM01	“Specify attributes specific to CICSplex SM” on page 50
WUISYSID		X	WU01	“Specify attributes specific to CICSplex SM” on page 50
XTRAQUAL	X		...	“Specify extra qualifiers” on page 35

## Specify the CICS Transaction Server temporary installation libraries

Specify the data set names you want to use for the two temporary libraries that will be used to install CICS Transaction Server. If you do not want to use the default names, record your values for the `TEMPLIB` and `LIB` parameters.

### `TEMPLIB` library\_name

This specifies the name of the temporary installation library that contains the skeleton installation jobs. Specify the name of the data set into which you copied `RELFILE(2)` from the distribution tape (in the topic *Copy RELFILE(2) from the Distribution Tape* in the *Program Directory*).

Also specify this name on the `SYSPROC` DD statement of the DFHISTAR job.

### `LIB` library\_name

Specifies the name of the installation output library to which the jobs generated by the DFHISTAR job are added.

## Specify the JOB parameters for installation jobs

Decide what parameters you want to use on the `JOB` statements of the CICS Transaction Server installation jobs, and specify them on the `JOB` parameter:

### `JOB` accounting\_information

Specifies the `JOB` statement and accounting information that you want substituted into the jobs generated by the DFHISTAR job. For example:

```
JOB //XXXXXXXX JOB 1,userid,MSGCLASS=A,MSGLEVEL=(1,1),
JOB //          CLASS=A,NOTIFY=userid
JOB /*JOBPARM SYSAFF=node1
JOB /*ROUTE PRINT node2.userid
```

### Note:

1. Do not change `XXXXXXXX` given in the sample `JOB` statement in the DFHISTAR job. This is the 8-character job name that is substituted by the DFHISTAR job. For example, for the installation job DFHIVPBT, the DFHISTAR job changes `XXXXXXXX` to DFHIVPBT.

2. Normal JCL rules for coding JOB statements apply to the JOB parameter.
3. If you want to add a TIME parameter to the CICS Transaction Server installation jobs, sample run times are given in “Run Times of the Installation Jobs” on page 56.
4. Delete (or comment out) extra lines of the JOB statement that you do not need.
5. Normal JCL rules apply when coding the JOB statement (for example, all lines except the last line must end in a comma).

## Specify the scope of the installation

Specify the scope of the CICS Transaction Server installation on the SCOPE parameter:

**SCOPE** ALL|BASE|POST

Specifies whether you want to generate all the CICS Transaction Server installation and post-installation jobs, or only the post-installation jobs. When installing CICS Transaction Server from the distribution tape, you would normally specify SCOPE ALL (the default). You would normally code the other options, if necessary, during the post-installation tasks.

**ALL**

Specifies that you want to generate all the CICS Transaction Server installation jobs and all the post-installation jobs.

**BASE**

Specifies that you want to generate only the installation jobs (DFHINST1 through DFHINST6, DFHIHFS0, DFHIHFS1, and DFHISMKD) that you use to install CICS Transaction Server from the distribution tape.

**POST**

Specifies that you want to generate only the post-installation jobs, that you can use to create the CICS Transaction Server data sets, and run the IVPs.

## Specify the type of JES to be used

Specify the type of job entry subsystem (JES) that you intend to use to install CICS Transaction Server on the JES parameter: This enables the DFHISTAR job to generate jobs with statements suitable for JES2 or JES3.

**JES** JES2|2|JES3|3

Specifies the release of JES that you are using. If you are using JES2, specify *JES2* or *2*. If you are using JES3, specify *JES3* or *3*.

## Specify the utilities to be used

Specify the utilities to be used to install CICS Transaction Server on the UTILITIES parameter:

**UTILITIES** *asmprog binder smpeprog copyutil*

Specifies the names of utility programs to be used when installing CICS Transaction Server elements and programs that it uses.

*asmprog*

is the program name of the assembler. Specify ASMA90, for High Level Assembler/MVS & VM & VSE, which is required.

*binder*

is the program name of the z/OS binder. Ensure that program IEWL references the z/OS program management binder.

*smpeprog*

is the program name of the SMP/E program. The IBM-supplied name is GIMSMP.

*copyutil*

is the program name of the data set copy utility program. The IBM-supplied name is IEBCOPY.

**Note:** The High Level Assembler must be either in the LINKLST concatenation **or** you must add a STEPLIB DD statement that points to the library containing the High Level Assembler in any jobs that invoke SMP/E.

## Specify the prefix of CICS Transaction Server jobs

Specify the prefix, as one through six characters, to be added to the jobs generated by the DFHISTAR job. This prefix overwrites the first characters of the job name. For example, PREFIX USERID changes the job name DFHINST1 to USERIDT1.

### **PREFIX prefix**

The 1- to 6-character prefix to be added to the CICS jobs generated by the DFHISTAR job.

## Specify the indexes of CICS Transaction Server data sets

Specify the high-level indexes for the CICS Transaction Server distribution, target, and SMP/E libraries allocated by the installation process.

### **DINDEX library\_prefix**

Assigns a high-level index to the CICS Transaction Server SMP/E distribution libraries allocated by the installation process.

The *library\_prefix* value must not be longer than 26 characters, and the leading character must be alphabetic. If you specify more than one level of index, the names must be separated by a period (for example, DINDEX C1CSTS32.TEST).

### **GINDEX library\_prefix**

Assigns a high-level index to the CICS Transaction Server SMP/E global libraries allocated by the installation process.

The *library\_prefix* value must not be longer than 26 characters, and the leading character must be alphabetic. If you specify more than one level of index, the names must be separated by a period (for example, GINDEX C1CSTS32.TEST).

### **LINDEX library\_prefix**

Assigns a high-level index to the SDFHLPA, SDFHLINK, SEYULINK and SEYULPA libraries allocated by the installation process. The *library\_prefix* value must be defined in the MVS Master Catalog.

The *library\_prefix* value must not be longer than 26 characters, and the leading character must be alphabetic. If you specify more than one level of index, the names must be separated by a period (for example, LINDEX SYS1. C1CSTS32. CICS.TEST).

### **TINDEX library\_prefix**

Assigns a high-level index to the CICS Transaction Server SMP/E target libraries (except for the SDFHLINK, SDFHLPA, SEYULINK, and SEYULPA target libraries) allocated by the installation process.

### **Note:**

1. The high-level index for the SDFHLINK and SDFHLPA libraries is defined by the LINDEX parameter.

- The high-level index for the data sets created by the jobs DFHCOMDS and DFHDEFDS is defined by the *dsindex* operand of the DSINFO parameter.

The *library\_prefix* value must not be longer than 26 characters, and the leading character must be alphabetic. If you specify more than one level of index, the names must be separated by a period (for example, TINDEX CICSTS32.TEST).

## Specify the CICS TS z/OS UNIX directories and data sets

The DFHISTAR job has parameters that enable you to customize the UNIX system services z/OS UNIX directories.

**Note:** When discussing z/OS UNIX Directory names, this topic and the *Program Directory* use different methods to show fixed and variable names, as follows:

Table 5.

This topic uses:	the Program Directory uses:	to show:
<i>/variable</i>	/@variable@	a variable value
<b>/constant</b>	/constant	a fixed value

### pathprefix

The name of an optional prefix to the CICS TS directory **/usr/lpp/cicsts**. For example:

*/example/usr/lpp/cicsts*

### ussdir

The name of the CICS TS directory under */pathprefix***/usr/lpp/cicsts**.

The full name is therefore */pathprefix***/usr/lpp/cicsts/ussdir**. **Ussdir** is a name that you can choose. The default for *ussdir* is the value of the TINDEX parameter in lowercase.

The default path is:

*/pathprefix***/usr/lpp/cicsts/cicsts32**

**Note:** The name of the UNIX System Services directory after the root directory (**/usr/lpp**) is always **/cicsts**.

### HFS0DSN

The dataset name of the file system to be mounted at directory */pathprefix***/usr/lpp/cicsts/cicsts32**.

Note that these directory names are fixed. The default is data set name OMVS.USR.LPP.CICSTS .

See “The DFHIHFS0 Job” on page 56 for details of the job that uses this parameter.

### HFS1DSN

The dataset name of the file system to be mounted at directory */pathprefix***/usr/lpp/cicsts/ussdir**, where *ussdir* is the name of the directory specified on the **ussdir** parameter in the DFHISTAR job. The default is data set name OMVS.USR.LPP.CICSTS.CICSTS32.

See “The DFHIHFS1 Job” on page 57 for details of the job that uses this parameter.



## HFSADSN

The dataset name of the z/OS UNIX equivalent of the SMP/E “additional target zone”, to be mounted at directory */pathprefix/usr/lpp/cicsts/ussdira*, where *ussdira* is the name of the directory specified on the **ussdira** parameter in the DFHISTAR job. The default data set name is OMVS.USR.LPP.CICSTS.CICSTS32.A.

See “Create Extra Sets of CICS Transaction Server Target Libraries (Optional)” on page 64 for details of the job that uses this parameter.

## Specify extra qualifiers

Specify extra qualifiers that can optionally be inserted into the data set name of the 'target', 'distribution' and 'additional' zone data sets respectively, and inserted before the last data set qualifier. For example, XTRAQUAL JDOE . . changes the name of the target zone libraries to the values set by TINDEX.CICSTS32.CICS.JDOE.SDFHLOAD.

XTRAQUAL . . .

Three qualifiers to be used by 'target', 'distribution' and 'additional' zone data sets. If a qualifier is not required, specify a period (.).

## Specify block sizes

Specify the block sizes to be used when allocating data sets during installation on the BLKFB80 and BLKU parameters:

**BLKFB80** {0|*blocksize*}

The block size to be used when allocating data sets that have a fixed block record format and record length of 80 bytes.

The IBM-supplied value in DFHISTAR is 0. You are recommended to leave this value specified as 0 to allow z/OS to determine the optimum block size for you.

**BLKU** {32760|*blocksize*}

The block size to be used when allocating data sets that have an **undefined** record length.

## Specify the disk unit for work data sets

Specify the UNIT parameter for the disk or disks on which work data sets are stored on the WORKUNIT parameter:

**WORKUNIT** **disktype**

a unit identifier.

## Specify SMS option for DASD allocations

Specify the extent to which you want to leave the allocation of CICS TS installation data sets to SMS. You can either let SMS handle all DASD allocation, or you can opt to use volume parameters to control the allocations that do not have to be SMS-managed.

**Note:** Some CICS TS data sets are installed in PDSE data sets. These are:

- The SMP/E SMPLTS data set.
- The distribution library, ADFJMOD.
- The target libraries SDFJLOAD and SDFJAUTH.

See the topic *DASD Storage Requirements* in the *Program Directory for CICS Transaction Server for z/OS* for information about these data sets.

## SMS Y|N

If you specify SMS Y, the VOLUME parameter is omitted from the generated installation jobs, and all data set allocations are handled by SMS.

If you specify SMS N, the VOLUME parameter is included on the generated installation jobs, and will be honored according to your SMS configuration. The VOLUME parameters used are those specified on the ADDTVOL, DEFVOL, DISTVOL, CMACVOL, OPTVOL, SMPVOL, and TARGVOL parameters.

## Specify disk volumes

If you intend to install CICS Transaction Server into disk space managed by the storage management subsystem (SMS)<sup>1</sup> component of MVS/DFP, you do not need to specify your own disk volumes; device assignment can be determined by SMS. In this case, proceed to “Specify SMP/E Zone Attributes” on page 41.

To allow you to make the best use of your disk space, you can specify your own disk volumes and device types to be used to install CICS Transaction Server. You can specify your own disk details on the following parameters:

### DEFVOL volume disktype

Defines the default disk on which the contents of the disk volumes CMACVOL, DISTVOL, OPTVOL, SMPVOL, and TARGVOL will reside if the appropriate parameter is not coded in the DFHISTAR job. For example, if you do not code the DISTVOL parameter, the CICS Transaction Server distribution libraries will reside on the disk defined by DEFVOL.

#### volume

is one of the following:

- The volume serial identifier, in the range 1 through 6 characters, of the default volume.
- A period (.) if all volumes other than CMACVOL and SMPVOL that are not specifically defined by the appropriate parameter of the DFHISTAR job will be put onto any available volume. The CMACVOL and SMPVOL volumes will be put onto the same volume as the library specified by the TEMPLIB parameter.

#### disktype

is the UNIT parameter of the volume.

If you omit the DEFVOL parameter altogether, all volumes that are not specifically defined by the appropriate parameter of the DFHISTAR job will be put onto the same volume as the library specified by the TEMPLIB parameter.

### DISTVOL volume disktype

Defines the disk on which the CICS Transaction Server distribution libraries will reside.

#### volume

is one of the following:

- The volume serial identifier, in the range 1 through 6 characters, of the volume on which the distribution libraries will reside.
- A period (.) if the CICS Transaction Server libraries are to be put onto any available volume.

---

1. For further information about installing system-managed storage, and about planning for and migrating storage to an SMS-managed environment, see the *MVS Storage Management Library: Storage Management Subsystem Migration Planning Guide*, SC26-4406.

**disktype**

is the UNIT parameter of the volume.

**Note:** If you omit the DISTVOL parameter, the distribution libraries will be put on the volume specified by the DEFVOL parameter. If the DEFVOL parameter is omitted, or if a period (.) is specified for its *volume* operand, the distribution libraries will be put onto any available volume.

**TARGVOL volume disktype**

Specifies details of the disk containing the CICS Transaction Server target libraries.

**volume**

is one of the following:

- The volume serial identifier, in the range 1 through 6 characters, of the volume on which the CICS Transaction Server target libraries are to reside.
- A period (.) if the CICS Transaction Server target libraries are to be put onto any available volume.

**disktype**

is the UNIT parameter for the volume.

**Note:** If you omit the TARGVOL parameter, the CICS Transaction Server target libraries will be put onto the volume specified by the DEFVOL parameter. If the DEFVOL parameter is omitted, or if a period (.) is specified for its *volume* operand, the CICS Transaction Server target libraries will be put onto any available volume.

**SMPVOL volume disktype**

Specifies the disk that contains the permanent, non-VSAM SMP/E data sets for CICS Transaction Server that are associated with global or distribution zones, and are therefore unique.

**volume**

is one of the following:

- The volume serial identifier, in the range 1 through 6 characters, of the volume on which the permanent non-VSAM SMP/E data sets are to reside.
- A period (.) if the permanent non-VSAM SMP/E data sets are to be put onto the same volume as the library specified by the TEMPLIB parameter.

**disktype**

is the UNIT parameter for the volume.

**Note:** If you omit the SMPVOL parameter, the permanent non-VSAM SMP/E data sets for CICS Transaction Server will be put on the volume specified by the DEFVOL parameter. If the DEFVOL parameter is omitted, or if a period (.) is specified for its *volume* operand, the data sets will be put onto the same volume as the library specified by the TEMPLIB parameter.

**OPTVOL volume disktype**

Specifies details of the disk onto which the optional source material is copied.

**volume**

is one of the following:

- The volume serial identifier, in the range 1 through 6 characters, of the volume on which the optional source material is to reside.
- A period (.) if the optional source material is to be put on any available volume.

**disktype**

is the UNIT parameter of the volume. This is needed only if *volume* is specified.

**Note:** If you omit the OPTVOL parameter, the optional source material will be put on the volume specified by the DEFVOL parameter. If the DEFVOL parameter is omitted, or if a period (.) is specified for its *volume* operand, the optional source material will be put onto any available volume.

**CMACVOL volume**

Defines the disk on which the VSAM KSDS, DFHMACD, will reside. This data set is used for the CICS Transaction Server messages facility (CICS-supplied transaction CMAC).

**volume**

is one of the following:

- The volume serial identifier, in the range 1 through 6 characters, of the volume on which the VSAM KSDS, DFHMACD, will reside
- A period (.) if the DFHMACD data set is to be put onto the same volume as the library specified by the TEMPLIB parameter.

**Note:** If you omit the CMACVOL parameter, the DFHMACD data set will be put onto the volume specified by the DEFVOL parameter. If the DEFVOL parameter is omitted, or if a period (.) is specified for its *volume* operand, the DFHMACD data set will be put onto the same volume as the library specified by the TEMPLIB parameter.

**When Are These Volumes Used?**

DFHISTAR Volume Parameter	Installing	Applying Service	Customizing	Assembling Resource Tables	Running CICS Transaction Server
SMPVOL	Yes	Yes	Yes	Yes	
DISTVOL	Yes	Yes	Yes		
TARGVOL	Yes	Yes	Yes	Yes	Yes
DZONECSI <sup>1</sup>	Yes	Yes	Yes		
TZONECSI <sup>1</sup>	Yes	Yes	Yes	Yes	
GZONECSI <sup>1</sup>	Yes	Yes	Yes	Yes	

<sup>1</sup> The entries for xZONECSI parameters are also for the associated xZONE parameters.

**During installation:** The RELFILE data sets on SMPVOL are needed during installation only.

**Applying Service or Customizing CICS Transaction Server:**

SMPVOL, DISTVOL, TARGVOL, DZONE, TZONE, and GZONE are needed whenever you apply service or customize your CICS Transaction Server programs.

SMPVOL and GZONE are needed whenever you apply service or customize your *alternative* libraries for use with the extended recovery facility.

**Assembling CICS Transaction Server Tables:**

SMPVOL, TARGVOL, TZONE, and GZONE are needed whenever you assemble your CICS Transaction Server tables.

SMPVOL and GZONE are needed whenever you assemble CICS Transaction Server tables for the second (alternate) CICS Transaction Server region.

**Running CICS Transaction Server:**

Only TARGVOL is needed to run CICS Transaction Server.

**Allocate space for CICS Transaction Server disk volumes**

Whether or not you use SMS-managed data sets, you still need enough disk space in which to create the CICS Transaction Server disk volumes.

The space required by the installation jobs on these volumes depends on the type of disk you intend to use. The number of cylinders required on the different types of DASD are given in Table 6. The size of the CICS Transaction Server distribution and target libraries are given in the topic *DASD Storage Requirements* in the *Program Directory* .

Table 6. DASD storage requirements for CICS Transaction Server

Identification	3380	3390
CICSTS32.TDFHINST	1	1
CICSTS32.XDFHINST	1	1
Relfile data sets on SMPVOL	325	300
SMP/E non-VSAM data sets on SMPVOL	26	25
DISTVOL	312	288
TARGVOL	749	678
DZONE	11	11
TZONE	11	11
GZONE	11	11
Total during installation	1447	1327
Total after installation	1122	1027

Allow up to 15% on the values in Table 6 for servicing requirements. Secondary allocations are 10% of the primary allocations.

If you intend to store other IBM software or your own application programs in these libraries, then you must modify the generated jobs accordingly.

**Specify Attributes of the Temporary SMP/E Work Data Sets**

You must define the attributes of the temporary SMP/E work data sets (SMPWRK1, SMPWRK2, SMPWRK4, and SMPWRK6): for the following CICS Transaction Server jobs: DFHINSTJ, DFHLPJPN, DFHLPUMD, and DFHSMPE. You define the attributes of those SMP/E data sets on the SMPWORK parameter:

The CICS Transaction Server jobs used to install CICS Transaction Server for z/OS have DD statements for the SMP/E data sets that they need to know about.

#### **SMPWORK disktype**

This is the UNIT parameter for the disk that is to contain the temporary SMP/E work data sets (SMPWRK1, SMPWRK2, SMPWRK4, and SMPWRK6) needed to install CICS Transaction Server.

If you specify a value for *disktype*, or omit the SMPWORK parameter altogether, //SMPWRKn DD statements are added to the following jobs generated by the DFHISTAR job:

- DFHINSTJ
- DFHLPUMD
- DFHSMPE

If you specify NO, a period (.), or a null string, CICS Transaction Server assumes that SMP/E knows about the temporary SMP/E work data sets. To define the attributes of the SMP/E work data sets, you must do one of the following:

- Provide appropriate DDDEFS for the temporary SMP/E work data sets.
- Have applied the SMP/E sample usermod (SMP0001) that contains superzap statements for updating the default attributes of the SMP/E data sets in the GIMMPDFT module.

The GIMMPDFT module, which is part of SMP/E, defines the default attributes of SMP/E data sets, and can be used to dynamically allocate data sets to be used by all zones. The usermod is in member GIMZPDFT in the SYS1.SAMPLIB library. You can use this usermod as a model, change it to meet your needs, or install it as supplied. For further information about the entries in the GIMMPDFT module and the sample entry values in the usermod SMP0001, see the *System Modification Program Extended: Reference* manual, SA22-7772.

#### **Note:**

1. The SMPWRK6 data set must not be allocated to Virtual I/O (VIO). If you specify a value for *disktype*, ensure that this cannot happen.

## **Specify Attributes of the Permanent SMP/E Data Sets**

Specify the attributes of the permanent SMP/E data sets on the following parameters:

#### **SMPPTS dsname**

Specifies the name of the SMP/E primary data set used to store temporarily PTF function SYSMODs or other fixes that are in RECEIVE or APPLY status; that is, PTF fixes that have not been rejected or accepted.

#### **SMPMTS dsname**

Specifies the name of the SMP/E macro temporary store (MTS) data set used to store updated versions of macros. Although required by SMP/E, this MTS data set is not used by CICS.

#### **SMPSTS dsname**

Specifies the name of the SMP/E source temporary store (STS) data set used to store updated versions of source elements. Although required by SMP/E, this STS data set is not used by CICS.

**SMPCDS dsname**

Specifies the name of the SMP/E saved control data set (SCDS) used to store old target zone entries that have been modified by inline JCLIN processing in a SYSMOD.

**SMPLTS dsname**

Specifies the name of the linkedit temporary (LTS) data set used with the CALLLIBS function. This dataset must always be an SMS-managed PDSE, whether or not Java is installed unless you have installed the PTFs for z/OS UNIX and PDSE support on non-managed SMS volumes listed in Flash10007 which can be found by links from:

<http://www.ibm.com/support/techdocs>

The CICS Transaction Server jobs that need to know the attributes of the SMP/E data sets have DD statements for them.

## Specify SMP/E Zone Attributes

Specify the attributes of the SMP/E distribution zone, global zone, target zone, and any additional target zones.

**Note:** The CICS-DB2 attachment facility contains modules named with the DSN prefix. Therefore, to prevent existing DB2 modules with the same DSNxxxxx names from being overwritten, you should not install CICS Transaction Server into the same target and distribution zones as DB2.

To specify SMP/E zone attributes, use the following parameters:

**GZONELOG dsname NEW|OLD**

Specifies details of the SMP/E log for the global zone CSI.

**dsname**

is the name of the global zone log.

**NEW|OLD**

Specifies whether an existing global zone log is to be used. If you specify NEW, any existing global zone log with the specified *dsname* is deleted, and a new global zone log is allocated. If you specify OLD, an existing global zone log is used.

**TZONELOG dsname NEW|OLD**

Specifies details of the SMP/E log for the target zone CSI.

**dsname**

is the name of the target zone log.

**NEW|OLD**

Specifies whether an existing target zone log is to be used. If you specify NEW, any existing target zone log with the specified *dsname* is deleted, and a new target zone log is allocated. If you specify OLD, an existing target zone log is used.

**DZONELOG dsname NEW|OLD**

Specifies details of the SMP/E log for the distribution zone CSI.

**dsname**

is the name of the distribution zone log.

**NEW|OLD**

Specifies whether an existing distribution zone log is to be used. If you

specify NEW, any existing distribution zone log with the specified *dsname* is deleted, and a new distribution zone log is allocated. If you specify OLD, an existing distribution zone log is used.

**GZONECSI cluster NEW|OLD volume disktype**

Specifies details of the global zone CSI.

**cluster**

is the VSAM cluster name, minus the qualifier '.CSI'.

**NEW|OLD**

Specifies whether an existing global zone CSI is to be used. If you specify NEW, any existing global zone CSI with the specified *cluster* name is deleted, and a new global zone CSI is allocated. If you specify OLD, an existing global zone CSI is used.

**volume**

is either the volume serial (volser) identifier for the volume on which the global zone CSI is to be allocated or a period (.) if the CSI is to be put on a volume determined by the CICS Transaction Server installation process.

**disktype**

is the UNIT parameter for the volume.

**TZONECSI cluster NEW|OLD volume disktype**

Specifies details of the target zone CSI.

**cluster**

is the VSAM cluster name, minus the qualifier '.CSI'.

**NEW|OLD**

Specifies whether an existing target zone CSI is to be used. If you specify NEW, any existing target zone CSI with the specified *cluster* name is deleted, and a new target zone CSI is allocated. If you specify OLD, an existing target zone CSI is used.

**volume**

is either the volume serial (volser) identifier for the volume on which the target zone CSI is to be allocated or a period (.) if the CSI is to be put on a volume determined by the CICS Transaction Server installation process.

**disktype**

is the UNIT parameter for the volume.

**DZONECSI cluster NEW|OLD volume disktype**

Specifies details of the distribution zone CSI.

**cluster**

is the VSAM cluster name, minus the qualifier '.CSI'.

**NEW|OLD**

specifies whether an existing distribution zone CSI is to be used. If you specify NEW, any existing distribution zone CSI with the specified *cluster* name is deleted, and a new distribution zone CSI is allocated. If you specify OLD, an existing distribution zone CSI is used.

**volume**

is either the volume serial (volser) identifier for the volume on which the distribution zone CSI is to be allocated or a period (.) if the CSI is to be put on a volume determined by the CICS Transaction Server installation process.



**disktype**

is the UNIT parameter for the volume.

**GZONE NEW|OLD options**

Specifies whether the global zone to be used already exists.

**NEW|OLD**

Specifies whether an existing global zone is to be used. The DFHISTAR job as supplied specifies NEW. Optionally change this to OLD if you want to use an existing global zone. If you specify OLD, CICS Transaction Server is installed into an existing SMP/E global zone.

**Note:** Specify NEW if you want to preserve your existing releases of CICS Transaction Server in their current SMP/E zones and install the new release in its own zones.

If you specify OLD, the existing SMP/E zones are used and any existing release of the product is deleted.

If you specify OLD, but specify NEW for the GZONECSI parameter, both parameters are assigned the NEW disposition.

**options**

Specifies the name of the SMP/E options (on the SET BOUNDARY command) to be used.

**TZONE zonename**

Specifies the name of the target zone.

**zonename**

is the name of the target zone to be used by SMP/E. This name must be unique to the target zone. It must not be longer than 7 characters, and the leading character must be alphabetic.

**DZONE zonename**

Specifies the name of the distribution zone.

**zonename**

is the name of the distribution zone to be used by SMP/E. This name must be unique within the global zone. It must not be longer than 7 characters, and the leading character must be alphabetic.

**SMP/E Zone and Zone Log Dispositions**

As supplied, the DFHISTAR job assumes that you are going to install CICS Transaction Server into new target and distribution zones. However you can specify a new or old global zone, and new or old zone logs by the disposition option NEW|OLD on the associated parameters of the DFHISTAR job. The disposition option NEW means that the DFHINST3 job deletes any existing zone or zone log with name specified before redefining it. For example, if you specify the parameter `GZONELOG CICSTS32.GZONE.SMPLOG NEW`

the DFHINST3 job deletes any existing SMP/E global zone log with the name `CICSTS32.GZONE.SMPLOG` before defining a new SMP/E global zone log with that name.

Further, if you specify different dispositions for a zone parameter and its associated zone log parameter, they are both given the default disposition NEW. This is to ensure that both a zone and its zone log have the same disposition.

If you intend to install CICS Transaction Server using one new CSI for all zones, you must specify the disposition **NEW** on all three CSI parameters of the DFHISTAR job. For example:

```
DZONE          DZONE
DZONECSI      CICSTS32.SMPZONE NEW CICSTS32 SYSALLDA
DZONELOG      CICSTS32.DZONE.SMPLOG NEW
GZONE         NEW CICSOPT
GZONECSI      CICSTS32.SMPZONE NEW CICSTS32 SYSALLDA
GZONELOG      CICSTS32.GZONE.SMPLOG NEW
TZONE        TZONE
TZONECSI      CICSTS32.SMPZONE NEW CICSTS32 SYSALLDA
TZONELOG      CICSTS32.TZONE.SMPLOG NEW
```

## Specify the High-level Qualifiers for SMP/E Data Sets

For each different high-level qualifier that you have specified for SMP/E zone CSIs, logs, and other SMP/E data sets, you must create an ALIAS definition in the master catalog before the data sets can be used.

## Specify the distribution tape device type

Specify the type of device that is to be used to load the CICS Transaction Server distribution tape on the TAPEUNIT parameter:

### TAPEUNIT devicetype

Specifies the device type to be used to read the distribution tape. Use 3480 for the 3480 tape cartridge, 3400-6 for the 6250 tape, or the unit names in use in your installation.

## Specify attributes of the CICS Transaction Server system data sets

Specify attributes of the CICS Transaction Server system data sets, to be created when you run the post-installation jobs DFHCOMDS, DFHDEFDS<sup>2</sup>, EYUCMADS, EYUWUIDS, and EYUCSYDS jobs, on the DSINFO parameter:

### DSINFO dsindex volume disktype qualifier

Defines the following attributes of CICS TS system data sets:

#### dsindex

Assigns a high-level index to all the data sets defined by the jobs, DFHCOMDS, DFHDEFDS, EYUCMADS, EYUWUIDS, and EYUCSYDS.

The leading character of *dsindex* must be alphabetic. *dsindex* can have one or two levels of index, but each level must be no longer than eight characters. If you specify more than one level of index, the names must be separated by a period (for example, CICSTS32.CICSHTC1).

#### volume

is the volume identifier of the volume.

#### disktype

is the UNIT parameter for the volume.

#### qualifier

is a partial qualifier added to the index for the data sets created by the jobs DFHCOMDS, DFHDEFDS, EYUCMADS, EYUWUIDS, and EYUCSYDS. You can specify a partial qualifier of up to four alphanumeric characters; these characters are appended to the characters CICS to make the qualifier. If you specify a period (.) no qualifier is used.

2. For more information about the post-installation jobs DFHCOMDS and DFHDEFDS jobs, see "Creating data sets common to all CICS regions, DFHCOMDS job" on page 220.

## Specify Attributes of any additional target Libraries

If you want to create extra copies of the CICS Transaction Server target libraries, specify the attributes of those libraries on the following parameters:

### **AINDEX library\_prefix**

Assigns a high-level index to the additional set of CICS target libraries copied by a version of the DFHINSTA job.

#### **Note:**

1. The high-level index for the additional SDFHLINK and SDFHLPA libraries is defined by the ALINDEX parameter.
2. The high-level index for the data sets created by the jobs DFHCOMDS and DFHDEFDS is defined by the *dsindex* operand of the DSINFO parameter.

The AINDEX value must be unique (for example, it must be different from the INDEX value), it must not be longer than 26 characters, and the leading character must be alphabetic. If you specify more than one level of index, the names must be separated by a period (for example, AINDEX CICS32.A.TEST).

### **ALINDEX library\_prefix**

Assigns a high-level index to the additional SDFHLPA and SDFHLINK libraries allocated by running a version of the DFHINSTA job.

The *library\_prefix* value must not be longer than 26 characters, and the leading character must be alphabetic. If you specify more than one level of index, the names must be separated by a period (for example, ALINDEX SYS1.CICS32.A.TEST)

### **AZONELOG dsname**

Specifies details of the SMP/E log for the additional target zone CSI.

#### **dsname**

is the name of the additional target zone log to be used by SMP/E.

### **AZONECSI cluster**

Specifies details of the additional target zone CSI. The CSI data set is created on the volume and unit specified by the ADDTVOL parameter.

#### **cluster**

is the VSAM cluster name, minus the qualifier **.CSI**.

### **AZONE zonename**

Specifies the name of the additional target zone, to be used for the set of CICS Transaction Server target libraries copied by a version of the DFHINSTA job.

#### **zonename**

is the name of the additional target zone to be used by SMP/E. This name must be unique to the target zone. It must not be longer than seven characters, and the leading character must be alphabetic.

### **ASMPSCDS dsname**

Specifies the name of the additional zone SMP/E SCDS data set.

#### **dsname**

is the name of the additional zone SMP/E SCDS data set.

### **ASMPMTS dsname**

Specifies the name of the additional zone SMP/E MTS data set.

#### **dsname**

is the name of the additional zone SMP/E MTS data set.

**ASMPSTS dsname**

Specifies the name of the additional zone SMP/E STS data set.

**dsname**

is the name of the additional zone SMP/E STS data set.

**ASMPSTS dsname**

Specifies the name of the additional zone SMP/E LTS data set.

**dsname**

is the name of the additional zone SMP/E LTS data set.

**ADDTVOL volume disktype**

Specifies the volume and unit type to contain all the additional zone data sets.

**volume**

is the volume serial identifier of the volume.

**disktype**

is the UNIT parameter for the volume.

**USSDIRA dsname**

Specifies the name of the UNIX System Services directory for the Additional Target zone. See job DFHINSTA.

The default is the value of the AINDEX parameter in lowercase.

The UNIX System Services directory path will start `/ussindex/cicsts/ussdira` where `ussindex` is the translated value of the USSINDEX parameter, and `ussdira` is the value of the USSDIRA parameter.

The default path will be: `/pathprefix/usr/lpp/cicsts/cicsts32.a`

**Note:** The name of the UNIX System Services directory after the root directory (`/usr/lpp/`) is always **cicsts**.

## Specify the Data Set Name of the SISPLoad Library

Specify the full data set name, up to 44 characters, of the library that contains ISPLINK (SISPLoad for ISPF Version 4 and above, or ISPLoad for ISPF version 3 and below). For example, `SISPLoad SYS1.USERID.SISPLoad` changes the SISPLoad library name to `SYS1.USERID.SISPLoad`. This library is accessed, as read-only, during the installation of CICS Transaction Server.

**SISPLoad dsname**

Up to 44 characters.

## Specify the Data Set Name of the CSSLIB Library

Specify the full data set name, up to 44 characters, of the CSSLIB library. For example, `CSSLIB SYS1.USERID.CSSLIB` changes the CSSLIB library name to `SYS1.USERID.CSSLIB`. This library is accessed, as read-only, during the installation of CICS Transaction Server.

**CSSLIB dsname**

Up to 44 characters.

## Specify the Data Set Name of the SCEECPP Library

Specify the full data set name, up to 44 characters, of the SCEECPP library. For example, `SCEECPP SYS1.USERID.SCEECPP` changes the SCEECPP library name to `SYS1.USERID.SCEECPP`. This library is accessed, as read-only, during the installation of CICS Transaction Server.

**SCEECPP dsname**  
Up to 44 characters.

## **Specify the Data Set Name of the SCEELKED Library**

Specify the full data set name, up to 44 characters, of the SCEELKED library. For example, SCEELKED SYS1.USERID.SCEELKED changes the SCEELKED library name to SYS1.USERID.SCEELKED. This library is accessed, as read-only, during the installation of CICS Transaction Server.

**SCEELKED dsname**  
Up to 44 characters.

## **Specify the Data Set Name of the SCEELKEX Library**

Specify the full data set name, up to 44 characters, of the SCEELKEX library. For example, SCEELKEX SYS1.USERID.SCEELKEX changes the SCEELKEX library name to SYS1.USERID.SCEELKEX. This library is accessed, as read-only, during the installation of CICS Transaction Server.

**SCEELKEX dsname**  
Up to 44 characters.

## **Specify the Data Set Name of the SCEEOBJ Library**

Specify the full data set name, up to 44 characters, of the SCEEOBJ library. For example, SCEEOBJ SYS1.USERID.SCEEOBJ changes the SCEEOBJ library name to SYS1.USERID.SCEEOBJ. This library is accessed, as read-only, during the installation of CICS Transaction Server.

**SCEEOBJ dsname**  
Up to 44 characters.

## **Specify the Data Set Name of the SCEESAMP Library**

Specify the full data set name, up to 44 characters, of the SCEESAMP library. For example, SCEESAMP SYS1.USERID.SCEESAMP changes the SCEESAMP library name to SYS1.USERID.SCEESAMP. This library is accessed, as read-only, during the installation of CICS Transaction Server.

**SCEESAMP dsname**  
Up to 44 characters.

## **Specify the Data Set Name of the SCLBSID Library**

Specify the full data set name, up to 44 characters, of the SCLBSID library. For example, SCLBSID SYS1.USERID.SCLBSID changes the SCLBSID library name to SYS1.USERID.SCLBSID. This library is accessed, as read-only, during the installation of CICS Transaction Server.

**SCLBSID dsname**  
Up to 44 characters.

## **Specify the Data Set Name of the SCSQLOAD Library**

Specify the full data set name, up to 44 characters, of the SCSQLOAD library. For example, SCSQLOAD SYS1.USERID.SCSQLOAD changes the SCSQLOAD library name to SYS1.USERID.SCSQLOAD. This library is accessed, as read-only, during the installation of CICS Transaction Server.

**SCSQLOAD dsname**  
Up to 44 characters.

## Specify the Data Set Name of the SCSQANLE Library

Specify the full data set name, up to 44 characters, of the SCSQANLE library. For example, SCSQANLE SYS1.USERID.SCSQANLE changes the SCSQANLE library name to SYS1.USERID.SCSQANLE. This library is accessed, as read-only, during the installation of CICS Transaction Server.

**SCSQANLE dsname**  
Up to 44 characters.

## Specify the Data Set Name of the SCSQCICS Library

If you want to run the WebSphere® MQ sample programs, specify the full data set name, up to 44 characters, of the SCSQCICS library. For example, SCSQCICS SYS1.USERID.SCSQCICS changes the SCSQCICS library name to SYS1.USERID.SCSQCICS. This library is accessed, as read-only, during the installation of CICS Transaction Server.

**SCSQCICS dsname**  
Up to 44 characters.

## Specify the Data Set Name of the SCSQAUTH Library

Specify the full data set name, up to 44 characters, of the SCSQAUTH library. For example, SCSQAUTH SYS1.USERID.SCSQAUTH changes the SCSQAUTH library name to SYS1.USERID.SCSQAUTH. This library is accessed, as read-only, during the installation of CICS Transaction Server.

**SCSQAUTH dsname**  
Up to 44 characters.

## Specify the installation directory for Java support

Specify the install location of the IBM Software Developer Kit for z/OS, Java 2 Technology Edition on z/OS UNIX. The install location is used in the CICS-supplied sample JVM profiles. The location in the JVM profiles determines which version of the SDK is used for Java support in a CICS region. You can switch versions at any time by changing this value after you have installed CICS.

The supplied value, java142/J1.4, points to the default installation directory for Version 1.4.2 of the IBM SDK for z/OS. If you have installed Version 5 of the SDK, and you want to set up the CICS region to use that version for Java support, the default installation directory for that version is java/J5.0.

CICS TS V3.2 supports only the 31-bit versions of the IBM SDK for z/OS, Java 2 Technology Edition, not the 64-bit versions.

**JAVADIR directory name**  
This parameter is appended to */pathprefix/usr/lpp/*, giving a full path name of */pathprefix/usr/lpp/javadir*.

## Specify the Data Set Name of the SDSNLOAD Library

Specify the full data set name, up to 44 characters, of the DB2 SDSNLOAD library. For example, SDSNLOAD SYS1.USERID.SDSNLOAD changes the SDSNLOAD library name to SYS1.USERID.SDSNLOAD. This library is accessed, as read-only, during the installation of CICS Transaction Server.

**Note:** The REXX for CICS element contains some modules that are link-edited against the DB2 load library, SDSNLOAD. If you do not have DB2 installed,

the DFHINST6 job fails because SDSNLOAD cannot be allocated to the job. To avoid this, define a dummy SDSNLOAD data set, with LRECL=0 and RECFM=U, and specify the name of this empty data set on the SDSNLOAD parameter.

**SDSNLOAD dsname**  
Up to 44 characters.

## **Specify the Data Set Name of the SEZARPCL and SEZACMTX libraries**

Specify the full data set names, up to 44 characters, of the SEZARPCL and SEZACMTX libraries. For example, SEZARPCL SYS1.USERID.SEZARPCL changes the SEZARPCL library to SYS1.USERID.SEZARPCL and SEZACMTX SYS1.USERID.SEZACMTX changes the SEZACMTX library name to SYS1.USERID.SEZACMTX. These libraries are accessed, as read-only, during the installation of CICS Transaction Server.

**SEZARPCL dsname**  
Up to 44 characters.

**SEZACMTX dsname**  
Up to 44 characters.

## **Specify the Data Set Name of the SCEECICS & SCEERUN libraries**

Specify the full data set names, up to 44 characters, of the SCEECICS & SCEERUN libraries. For example, SCEECICS SYS1.USERID.SCEECICS changes the SCEECICS library to SYS1.USERID.SCEECICS and SCEERUN SYS1.USERID.SCEERUN changes the SCEERUN library name to SYS1.USERID.SCEERUN. These libraries are accessed, as read-only, during the installation of CICS Transaction Server.

**SCEECICS dsname**  
Up to 44 characters.

**SCEERUN dsname**  
Up to 44 characters.

## **Specify the Data Set Name of the SCEERUN2 & SCEELIB libraries**

Specify the full data set names, up to 44 characters, of the SCEERUN2 & SCEELIB libraries. For example, SCEERUN2 SYS1.USERID.SCEERUN2 changes the SCEERUN2 library to SYS1.USERID.SCEERUN2 and SCEELIB SYS1.USERID.SCEELIB changes the SCEELIB library name to SYS1.USERID.SCEELIB. These libraries are accessed, as read-only, during the installation of CICS Transaction Server.

**SCEERUN2 dsname**  
Up to 44 characters.

**SCEELIB dsname**  
Up to 44 characters.

## **Specify the Data Set Name of the SCSFMOD0 and SIXMEXP libraries**

Specify the full data set names, up to 44 characters, of the SCSFMOD0 and SIXMEXP libraries. For example, SCSFMOD0 SYS1.SCSFMOD0 changes the SCSFMOD0 library to SYS1.SCSFMOD0. These libraries are accessed, as read-only, during the installation of CICS Transaction Server.

**SCSFMOD0 dsname**  
Up to 44 characters.

**SIXMEXP dsname**  
Up to 44 characters.

## Specify the Data Set Name of the SCEEBND2 library

Specify the full data set name, up to 44 characters, of the SCEEBND2 library. For example, SCEEBND2 SYS1.USERID.SCEEBND2 changes the SCEEBND2 library to SYS1.USERID.SCEEBND2. This library is accessed, as read-only, during the installation of CICS Transaction Server.

**SCEEBND2 dsname**  
Up to 44 characters.

## Specify Log stream and Log stream Structure Attributes.

Specify attributes of the CICS Transaction Server log streams and the coupling facility structures that you will use when you run the post-installation jobs DFHILG1, DFHILG2, DFHILG3 and DFHILG4<sup>3</sup> jobs, on the LOGGER-INFO parameter:

**LOGGER-INFO strsfx logsz shuntsz jnlasz gensz sysname loghlq logmodel**  
Defines the following attributes of CICS Transaction Server system data sets:

### **strsfx**

The last part of the coupling facility structure names, can be any three characters allowed in a structure name. Default 001. Used in DFHILG1, DFHILG2, DFHILG3 and DFHILG4.

### **logsz**

The Avgbufsize for system log streams in the LOG\_DFHLOG\_strsfx structure. Default 500. Used in DFHILG1.

### **shuntsz**

The Avgbufsize for shunted system log streams in the LOG\_DFHSUNT\_strsfx structure. Default 4096. Used in DFHILG1.

### **jnlasz**

The Avgbufsize for unforced user journal log streams in the LOG\_USERJRNL\_strsfx structure. Default 64000. Used in DFHILG1.

### **gensz**

The Avgbufsize for forced user journal log streams and forward recovery log streams in the LOG\_GENERAL\_strsfx structure. Default 2048. Used in DFHILG1.

### **sysname**

The MVS system name used to create model log streams for DFHLOG and DFHSUNT. Default MVSX. Used in DFHILG1.

### **loghlq**

The first qualifier of the model name for general logs and DFHLGLOG. Used in DFHILG3 and DFHILG4.

### **logmodel**

The second qualifier of the model name for general logs. Used in DFHILG3.

## Specify attributes specific to CICSplex SM

DFHISTAR attributes that are specific to CICSplex SM.

---

3. For more information about the post-installation jobs DFHILG1, DFHILG2, DFHILG3 and DFHILG4 jobs, see Chapter 27, "Defining the logger environment for CICS journaling," on page 169.



The following CICSplex SM specific attributes are used to customize the post-installation JCL for CICSplex SM. This JCL is used in the installation verification procedure for CICSplex SM, as described in Chapter 33, “CICSplex SM verification,” on page 361.

All attributes specific to CICSplex SM have defaults. If you do not wish to use CICSplex SM, you can run DFHISTAR without providing overrides for any of the following CICSplex SM specific parameters:

**CMASNAME {CMAS01 | *name*}**

The 1- to 8-character name to be allocated to a CMAS. The name can contain alphabetic, national, and numeric characters. However, the first character must be alphabetic or national.

The name of a CMAS must be unique within the CICSplex SM environment. It should not be the same as the name of another CMAS, a CICSplex, a CICS system, or a CICS system group.

**CMSSYSID value**

Specifies the 4-character system identifier of the CMAS. This identifier can contain alphabetic, national, and numeric characters. It must match the SYSIDNT system initialization parameter for the CMAS. The default is CM01.

**CSYSYSID value**

Specifies the 1- to 4-character system identifier for the managed CICS system. This identifier can contain alphabetic, national, and numeric characters. The default is CS01.

**CSYSNAME {CSYS01 | *name*}**

Specifies the 1- to 8-character name to be allocated to a MAS. The name can contain alphabetic, national, and numeric characters. However, the first character must be alphabetic or national.

The name of a MAS must be unique within the CICSplex SM environment. It should not be the same as the name of another MAS, a CICSplex, a CICS system, or a CICS system group.

**CSYSPLEX value**

Specifies the 1- to 8-character name to be allocated to a CICSplex of managed systems. This identifier can contain alphabetic, national, and numeric characters. The default is CSYPLX01.

The name of a CICSplex must be unique within the CICSplex SM environment. It should not be the same as the name of another CICSplex, a CICS system, or a CICS system group.

**NEWDREP dsname**

A new data repository that is being used by CICSplex SM.

**dsname**

Is the VSAM cluster name of the existing data repository.

The new CICS TS for z/OS, Version 3.2 data repository will have the name:

`dsinfo.EYUDREP.masname`

where:

**dsinfo**

Is the index specified with the DSINFO parameter.

**masname**

Is the name specified with the CMASNAME parameter.

Use a period to have an empty data repository created for CICS TS for z/OS, Version 3.2.

**OLDDREP dsname**

An existing data repository that is being used by a previous release of CICSplex SM. The records in the existing data repository are migrated to a new data repository for CICS TS for z/OS, Version 3.2. The existing data repository is not modified. If you do not specify this parameter, a new data repository is created.

**dsname**

Is the VSAM cluster name of the existing data repository.

The new CICS TS for z/OS, Version 3.2 data repository will have the name:

`dsinfo.EYUDREP.cmasname`

where:

**dsinfo**

Is the index specified with the DSINFO parameter.

**cmasname**

Is the name specified with the CMASNAME parameter.

Use a period to have an empty data repository created for CICS TS for z/OS, Version 3.2.

**TCPIPHST**

Specifies the 1- to 8-character name allocated to the TCP/IP host name for the WUI server. The default is XXXXXXXX.XXXXXXXX.XXXXXXXX.XXXXXXXX.

**TCPIPRT**

Specifies the 1- to 8-character identifier allocated to the TCP/IP port number for the WUI server. The identifier can contain numeric characters only, in the range 1 to 65535. The default is 12345.

**TIMEZONE code**

Specifies the time zone assigned to the data repository. This must be a single alphabetic character in the range B through Z. See *CICSplex System Manager Administration* for more information about setting the time zone. The default is B.

**WUI YES|NO**

Specifies whether to create a WUI CICSplex. This is ignored if the OLDDREP parameter is specified. The default is YES if OLDDREP is not specified.

**YES**

Create a WUI CICSplex.

**NO** Do not create a WUI CICSplex.

**WUINAME name**

Specifies the 1- to 8-character name allocated to a WUI. The name can contain alphabetic, national, and numeric characters. However, the first character must be alphabetic or national. The default is WUINCM01.

**WUIPLEX name**

Specifies the 1- to 8-character name allocated to a WUI CICSplex. The name can contain alphabetic, national, and numeric characters. However, the first character must be alphabetic or national. This is ignored if the OLDDREP parameter is specified. The default is created from the characters WUIP,

followed by the CMSSYSID. For example, using the default CMSSYSID, CM01, the default WUIPLEX name would be WUIPCM01.

**WUISYSID name**

Specifies the 1- to 4-character name allocated to a WUI system identifier. The name can contain alphabetic, national, and numeric characters. However, the first character must be alphabetic or national.

---

## Create RACF Profiles for the CICS Transaction Server Data Sets

Liaise with your Security Administrator to create appropriate RACF profiles for the CICS Transaction Server data sets, as described in the *CICS RACF Security Guide*.

At this stage, you need authority to access only the data set qualifiers specified on the TEMPLIB, LIB, and INDEX parameters. (The DFHISTAR job uses a temporary sequential data set, with the high-level qualifier specified on the INDEX parameter, to resolve the parameters to be substituted into the jobs being tailored.) However, it is worth coordinating the access authority for all the CICS Transaction Server data sets at the same time.

---

## Run the DFHISTAR Job

When you have edited the DFHISTAR job with the values of installation parameters for your CICS Transaction Server environment, save it.

When you are ready to tailor the skeleton jobs, submit the DFHISTAR job.

In addition to the UNIX system services z/OS UNIX jobs, there are three other file system-related members supplied in SDFHINST. These are DFHBPXP0, DFHBPXP1, and DFHBPXPA.

**DFHBPXP0**

This member contains a MOUNT command for inclusion in a BPXPRMxx member of the SYS1.PARMLIB dataset. The MOUNT command applies to the dataset specified in the HFS0DSN parameter of the DFHISTAR job to be mounted at directory */pathprefix/usr/lpp/cicsts*.

Add this mount command to the BPXPRMxx PARMLIB member after you have run the DFHIHFS0 job.

**DFHBPXP1**

This member contains a MOUNT command for inclusion in a BPXPRMxx member of the SYS1.PARMLIB dataset. The MOUNT command applies to the dataset specified in the HFS1DSN parameter of the DFHISTAR job to be mounted at directory */pathprefix/usr/lpp/cicsts/ussdir*, where *ussdir* is the name of the directory specified in the **ussdir** parameter in the DFHISTAR job

Add this mount command to the BPXPRMxx PARMLIB member after you have run the DFHIHFS1 job.

**DFHBPXPA**

This member contains a MOUNT command for inclusion in a BPXPRMxx member of the SYS1.PARMLIB dataset. The MOUNT command is for the dataset specified in the HFSADSN parameter of the DFHISTAR job to be mounted at directory */pathprefix/usr/lpp/cicsts/ussdira*, where *ussdira* is the name of the directory specified in the **ussdira** parameter in the DFHISTAR job.

Add this mount command to the BPXPRMxx PARMLIB member after you have run the DFHIHFS1A job.

When the DFHISTAR job has run, the jobs listed in the topic *Skeleton jobs in RELFILE(2) copied to the TDFHINST library* in the *Program Directory for CICS Transaction Server for z/OS* (apart from the DFHISTAR job) are tailored to your CICS Transaction Server environment and added to the library that you specified on the LIB parameter of the DFHISTAR job (by default, the CICSTS32.XDFHINST library). If necessary, the DFHISTAR job creates the library specified on the LIB parameter.

**The highest expected return code is 0.**

## Check the Output from the DFHISTAR Job

Check the output from the DFHISTAR job, and if needed, edit and submit the DFHISTAR job again.

The DFHISTAR job produces a job log and, if necessary, an error code:

- The output job log lists the values that were actually used for the parameters of the DFHISTAR job.
- If any error occurs when running the DFHISTAR job, an error code of 4 or 12 is returned. For error code 4, the skeleton jobs are tailored and added to the CICSTS32.XDFHINST library. For error code 12, the skeleton jobs are not tailored or copied. To resolve the cause of either error code, examine the output job log and, if necessary, edit and submit the DFHISTAR job again.

You can run the DFHISTAR job any number of times to alter the attributes of the jobs that it creates.

When running the DFHISTAR job after the first time, you can select specific jobs to be created, by using the SCOPE or SELECT parameter:

### **SCOPE ALL|BASE|POST**

Specifies whether you want to generate all the CICS Transaction Server installation and post-installation jobs, or only the post-installation jobs. When installing CICS Transaction Server from the distribution tape, you would normally specify SCOPE ALL (the default). You would normally code the other options, if necessary, during post-installation tasks.

#### **ALL**

Specifies that you want to generate all the CICS Transaction Server installation jobs and all the post-installation jobs.

#### **BASE**

Specifies that you want to generate only the six installation jobs DFHINST1 through DFHINST6 that you can use to install CICS Transaction Server from the distribution tape.

#### **POST**

Specifies that you want to generate only the post-installation jobs, that you can use to create the CICS Transaction Server data sets, and run the IVPs.

### **SELECT jobname newname**

Specifies the new name for a copy of a post-installation job to be generated when you run the DFHISTAR job. You can specify several SELECT parameters to select several post-installation jobs to be regenerated in one run of the DFHISTAR job. The SELECT parameter overrides the POST parameter; that is, if you use the SELECT parameter in the DFHISTAR job, only those jobs specified by SELECT are generated.

**Note:** If you are using the SELECT parameter to generate copies of the post-installation jobs for a new CICS Transaction Server region, you should also change the DSINFO parameter to specify details of the data sets for the new CICS region.

For example, to create copies of the jobs DFHDEFDS and DFHIVPOL for the CICS Transaction Server region CICSINS you could specify the DSINFO and SELECT parameters of the DFHISTAR job:

```
DSINFO userid.CICSTS32 H3P061 SYSALLDA INS
SELECT DFHDEFDS INSDEFDS
SELECT DFHIVPOL INSIVPOL
```

If you then run the DFHISTAR job, it would create the INSDEFDS job as a copy of the DFHDEFDS job, and the INSIVPOL job as a copy of the DFHIVPOL job, substituting the values that you specified on DSINFO into the new jobs.

You could then change the DSINFO and SELECT parameters, and run the DFHISTAR job to create other copies of the post-installation jobs for another CICS Transaction Server region.

---

## Check that You Are Ready to Run the Installation Jobs

Check that you are now ready to run the installation jobs:

1. Check the names of the data sets to be created by these jobs, because any existing data sets with those names are deleted by the installation jobs. If you want to keep an existing data set with a name specified in one of the installation jobs, you must change the name to be used for the new data set. For example, for the installation parameter *DZONECSI dsname NEW* the data set *dsname* is deleted and a new distribution zone CSI called *dsname* is allocated.
2. The CICS Transaction Server-supplied installation JCL will install CICS Transaction Server for z/OS into new target and distribution zones. If you want to install CICS Transaction Server into existing target and distribution zones, you must modify the DFHINST3 job.

**Caution:** *If you intend to use an existing target or distribution zone that contains an earlier release of CICS Transaction Server, be aware that any earlier release of CICS Transaction Server will be cleared before being replaced by CICS Transaction Server.*

3. If you intend to install CICS Transaction Server using both existing and new CSIs, any new CSIs must have the same control interval size as the existing CSIs.

If your existing CSIs do not have a control interval size of 4096 bytes, you must edit the DFHINST3 job (before running it) to change the CONTROLINTERVALSIZE(4096) parameter on the commands used to create the VSAM data sets for the new CSIs, to specify the same control interval size as the existing CSIs.

For further information about considerations for allocating CSI data sets, see the *System Modification Program Extended: Reference* manual, SA22-7772.

4. Ensure that you have appropriate RACF authority for the CICS Transaction Server data sets. For more information, see your Security Administrator and the *CICS RACF Security Guide*.

---

## Run the Installation Jobs

### OMVS requirement

Before running the installation jobs:

- Ensure the MVS image was IPLed with OMVS in full-function mode.
- Ensure the userid under which you are running the jobs has superuser authority

After you have run the DFHISTAR job to create the installation jobs, submit those jobs in sequence to install CICS Transaction Server. This section describes the CICS Transaction Server installation jobs, and gives considerations that may affect how you use them.

The CICS Transaction Server jobs are in the CICSTS32.XDFHINST library as a result of running the DFHISTAR job, which you copied from the distribution tape, as described in the topic *Copy RELFILE(2) from the Distribution Tape in the Program Directory for CICS Transaction Server for z/OS*.

These jobs should be run one at a time. Before you run a job, read the information about it (starting with section “The DFHIFHS0 Job.”

After you have run a job, check its output before proceeding to the next job. If a job terminates abnormally, find out why it failed (the job log lists the error messages produced on each run). Correct the error, and then proceed as advised in the job description. In any case, do not attempt to run the next job until the previous job has run successfully.

## Run Times of the Installation Jobs

To give you an idea what run times to expect, we ran the installation jobs on an IBM 9672 RX5 using a single LPAR. The run times were:

Job	Processor Time	Elapsed Time
DFHIFHS0	1 second	7 seconds
DFHIFHS1	1 second	7 seconds
DFHISMKD	1 second	1 second
DFHINST1	3 seconds	30 seconds
DFHINST2	3 seconds	30 seconds
DFHINST3	1 second	15 seconds
DFHINST4	1 second	15 seconds
DFHINST5	1 minute	11 minutes
DFHINST6	24 minutes	49 minutes

These time values are suitable to run the installation jobs on an IBM 9672 RX5 or bigger system. If you have a system smaller than an IBM 9672 RX5, you may need to review these values.

## The DFHIFHS0 Job

This job:

- Creates the file system specified on the HFS0DSN parameter of the DFHISTAR job

- Creates the **cicsts** directory at */pathprefix/usr/lpp/*
- Mounts the file system at directory */pathprefix/usr/lpp/cicsts*
- Changes the permission settings for the **/cicsts** directory to:
  - Owner=RWX
  - Group=RWX
  - Other=R-X
 (In octal form: 775)
 

Where:

  - R equates to Read
  - W equates to Write
  - X equates to Execute
  - - equates to no permission

**Note:**

1. DFHIHFS0 only ever needs to be run *once*.
2. RACF ALTER ACCESS to the OMVS data sets must be granted before running DFHIHFS0.
3. The **/cicsts** directory is common to all releases of CICS TS from 1.3 onwards.
4. The **/cicsts** directory contains only directories, each being a mount point.
5. CICS requires the MOUNT issued by DFHIHFS0 to access files stored in the file system, but the MOUNT command is lost when you re-IPL MVS. SDFHINST member DFHBPXP0 contains a MOUNT command for */pathprefix/usr/lpp/cicsts*. Copy this command into a BPXPRMxx member of the SYS1.PARMLIB dataset to ensure the mount is restored when MVS is IPLed.
6. All steps of DFHIHFS0 must end with return code zero for the job to be successful.

## The DFHIHFS1 Job

This job:

- Unmounts the file system at directory */pathprefix/usr/lpp/cicsts/ussdir* to allow the job to be rerun, and if necessary forces return code zero.
- Deletes from */pathprefix/usr/lpp/cicsts* the directory defined by the *lussdir* parameter of the DFHISTAR job. This is to allow the job to rerun, and if necessary forces return code zero.
- Deletes the file system specified in the HFS1DSN parameter of the DFHISTAR job to allow the job to rerun, and if necessary forces return code zero.
- Creates the file system specified in the HFS1DSN parameter of the DFHISTAR job.
- Creates the *lussdir* directory at */pathprefix/usr/lpp/cicsts*, where *lussdir* is the name of the directory specified on the **ussdir** parameter.
- Mounts the file system at directory */pathprefix/usr/lpp/cicsts/ussdir*
- Changes the permission settings for the *lussdir* directory to 775.

All steps of DFHIHFS1 must end with return code zero for the job to be successful.

CICS requires the MOUNT issued by DFHIHFS1 to access files stored in the z/OS UNIX file system, but the MOUNT command is lost when you re-IPL MVS.

SDFHINST member DFHBPXP1 contains a MOUNT command for */pathprefix/usr/lpp/cicsts/ussdir*. Copy this command into a BPXPRMxx member of the SYS1.PARMLIB dataset to ensure the mount is restored when MVS is IPLed.

## The DFHISMKD Job

This job creates the UNIX System Services directories.

This job **must** be run before any of the other installation jobs.

**The highest expected return code is 0.**

## The DFHINST1 Job

This job allocates and catalogs CICS Transaction Server distribution and target libraries.

To ensure that this job can be rerun, it deletes (and uncatalogs) the data sets that are allocated in the second step of the job.

If the DFHINST1 job terminates abnormally, examine the job log to determine the cause, correct the problem, then rerun the job.

**The highest expected return code is 0.**

## The DFHINST2 Job

This job allocates the CICS Transaction Server RELFILE data sets. If you run the DFHINST2 job now, you ensure that enough space has been allocated to the RELFILE data sets to allow the DFHINST5 job to complete.

To ensure that the job can be rerun, it deletes (and uncatalogs) the data sets (if they exist) that it allocates later.

If the DFHINST2 job terminates abnormally, examine the job log to determine the cause, correct the problem, then rerun the job.

**The highest expected return code is 0.**

### DFHINST3 and DFHINST4 jobs.

DFHINST3 is a job that should be run if you have decided to allocate new SMP/E zones.

Be aware that the default specified in DFHISTAR is for the allocation of new zones. This is recommended, to ensure that existing releases of CICS Transaction Server are not deleted.

DFHINST4 is a job that will prime new SMP/E zones created in DFHINST3.

## The DFHINST3 Job

This job allocates the CICS Transaction Server SMP/E data sets.

### Before you run the DFHINST3 job

If you intend to install CICS Transaction Server using both existing and new CSIs, any new CSIs must have the same control interval size as the existing CSIs.



**Caution: If you intend to use an existing target or distribution zone that contains an earlier release of CICS Transaction Server elements, be aware that any earlier release of CICS will be cleared before being replaced by CICS Transaction Server.**

If your existing CSIs do not have a control interval size of 4096 bytes, you must edit the DFHINST3 job (before running it) to change the CONTROLINTERVALSIZE(4096) parameter on the commands used to create the VSAM data sets for the new CSIs, to specify the same control interval size as the existing CSIs.

For further information about considerations for allocating CSI data sets, see the *System Modification Program Extended: Reference* manual, SA22–7772.

To ensure that the job can be rerun, it deletes (and uncatalogs) the data sets (if they exist) that it allocates later.

This job also sets up the global, target, and distribution zones, depending on the parameters that you specified to the DFHISTAR job, as follows:

1. If you specified NEW for GZONE, the global zone is deleted and redefined.
2. The distribution zone is deleted and redefined.
3. The target zone is deleted and redefined.
4. Member GIMZPOOL from SYS1.MACLIB is REPROed into the zones redefined in the previous steps.
5. If you specified OLD for GZONE, the entries for the DZONE and TZONE names are removed from the global zone.

If the DFHINST3 job terminates abnormally, examine the job log to find the cause, correct the problem, then rerun the job.

**The highest expected return code is 0.**

## The DFHINST4 Job

This job primes the global zone, target zone, and distribution zone.

### Before you run the DFHINST4 job

If you did not run the DFHINST2 job, increase the DSSPACE values in the DFHINST4 job, before submitting it.

If the DFHINST4 job terminates abnormally, examine the job log to determine the cause, correct the problem, then repeat all jobs, beginning with DFHINST1. This avoids SMP/E space problems, and consequent X37 abends, during reruns of these SMP/E jobs.

**The highest expected return code is 0, if you install into new zones and 8 if you are installing into existing zones.**

## The DFHINST5 Job

This job RECEIVES the CICS Transaction Server software from the distribution tape into the RELFILE data sets created by the DFHINST2 job. It is the only installation job (apart from the initial IEBCOPY job) that requires the distribution tape to be mounted.

If the DFHINST5 job terminates abnormally, examine the job log to determine the cause, correct the problem, then repeat all jobs, beginning with DFHINST1. This avoids SMP/E space problems, and consequent X37 abends, during reruns of these SMP/E jobs.

**The highest expected return code is 0.**

## The DFHINST6 Job

This job performs the SMP/E APPLY and ACCEPT functions needed to install CICS Transaction Server into the target and distribution libraries respectively.

### Before you run the DFHINST6 job

- Ensure the DB2 SDSNLOAD library is available for this job. Without SDSNLOAD, DFHINST6 fails; see “Specify the Data Set Name of the SDSNLOAD Library” on page 48

If you run DFHINST6 with an empty SDSNLOAD library, the APPLY step completes with return code 4, and the return code from the binder is 8. The REXX for CICS modules, CICSQL and CICDB2, which interface with DB2, are stored in SCICLOAD without the required DB2 routines and are not usable.

- **DFHINST6 must be run on the same MVS image on which the z/OS UNIX file system is installed (unless you are set up for MVS sharing).**

If you have modified the other installation jobs (for example, to use existing libraries and therefore existing target and distribution zones), consider splitting the DFHINST6 job to do APPLY CHECK, APPLY, ACCEPT CHECK, and ACCEPT functions as four separate jobs.

The DFHINST6 job is the longest running of all the installation jobs (see “Run Times of the Installation Jobs” on page 56), and produces a large amount of printed output. The region size for the DFHINST6 job is currently set to 'REGION=0M'. This is because this job requires more memory than the other install jobs. You may need to adjust your JES parameters (for example, with a JES2 /\*JOBPARM LINES=99 statement) to avoid a system abend 722.

**This job gives a return code of 4 when all is well. (See the “GIM23903W - LINK SUCCESSFUL . . ” message, listed in the report that is output by the apply job.) DFHINST6 job may issue messages GIM23903W and GIM23913W depending on the execution environment of the installer. Both these messages are acceptable.**

The binder produces IEW2454W messages during the APPLY stage for unresolved external references while some CICS Transaction Server load modules are being link-edited during installation, giving return code 4. You may also receive numerous IEW2646W and IEW2651W messages, which are conflicts with user-specified RMODE and AMODEs respectively. You can ignore these IEWxxxx messages, which are output for component object modules of executable CICS Transaction Server load modules.

Messages IEW2689W, IEW2470E, IEW2648E and IEW2303E might be displayed, and can also be ignored.

When you have run the DFHINST6 job, you should see the following SMP/E message in the output from the job:

```
GIM20502I GIMSMP PROCESSING IS COMPLETE - THE HIGHEST RETURN CODE WAS 04 -
```

This SMP/E message can be ignored.

If any other SMP/E messages appear, see the *SMP/E: Messages & Codes* manual for guidance information about their meaning, and take the appropriate action.

If the DFHINST6 job terminates abnormally, examine the job log to determine the cause, correct the problem, then repeat all the jobs, beginning with DFHINST1. This avoids SMP/E space problems, and consequent X37 abends, during reruns of these SMP/E jobs.

**Note:** If the DFHINST6 job fails and you are using an existing global zone (that is, you specified the GZONE parameter of the DFHISTAR job with the disposition parameter OLD), you must first REJECT the CICS Transaction Server base-level function SYSMOD before rerunning the DFHINST1 job. When you rerun the installation jobs, some steps that were successfully completed in the previous run will produce return codes with a value of '8'.

## The DFHIJVMJ Job

The DFHIJVMJ job creates the customized sample JVM profiles, JVM properties files and Java 2 security policy.

DFHIJVMJ reads the supplied JVM profiles and JVM properties files, and the security policy `dfjejbpl.policy`, from member DFHJVMEV in the partitioned dataset SDFHENV. It replaces the symbols `&CICS_HOME` and `&JAVA_HOME` in the files with the values you specify on the USSDIR and JAVADIR parameters in the DFHISTAR installation job. (The extra // characters on each side of the symbols in the supplied files are removed during symbol substitution.)

The customized JVM profiles, JVM properties files and Java 2 security policy are then written as z/OS UNIX files in the directories listed in “Verifying Java components checklist” on page 236.

---

## Check the Output from the Installation Jobs

When you have successfully run all of the installation jobs described in this chapter, CICS Transaction Server will have been loaded.

You now have CICS Transaction Server installed on your DASD. Back up the volume on which CICS Transaction Server resides. This avoids the need to re-run the installation jobs if any errors occur during customization later.

---

## What next?

You should next copy the CICS Transaction Server procedures into a cataloged procedure library, load any CICS features that you have, and tailor the CICS Transaction Server to your needs. For information about copying the CICS Transaction Server procedures, and loading the CICS Transaction Server features, see the following topics:

- “Copy the CICS Transaction Server procedures into a procedure library” on page 62
- “Create Extra Sets of CICS Transaction Server Target Libraries (Optional)” on page 64
- “Load the CICS Transaction Server Source Material (Optional)” on page 66
- “Load Other Optional Features” on page 66

. For information about tailoring CICS Transaction Server to your needs, see “Tailoring the CICS-supplied skeleton jobs” on page 215.

## Copy the CICS Transaction Server procedures into a procedure library

CICS Transaction Server supplies the procedures listed in Table 7.

Table 7. CICS Transaction Server-supplied procedures

Procedure	Description
DFHASMVS	Assembles some CICS Transaction Server programs and user-written assembler language programs.
DFHAUPL	Assembles and link-edits CICS Transaction Server control tables, and makes the assembly and link-edit information available to SMP/E. <b>Note:</b> DFHAUPL is installed in SDFHINST.
DFHBMSU	Runs the BMS load module disassemble utility program, DFHBMSUP.
DFHEBTAL	Translates, assembles, and link-edits assembler application programs using EXEC DLI commands in a batch environment under Language Environment.
DFHEITAL	Translates, assembles, and link-edits assembler application programs using the command-level interface.
DFHEXTAL	Translates, assembles, and link-edits assembler application programs using the external CICS Transaction Server interface.
DFHLNKVS	Link-edits CICS Transaction Server programs and application programs.
DFHMAPS	Prepares physical and symbolic maps.
DFHMAPT	Prepares physical and symbolic maps for C++.
DFHSMPE	Executes SMP/E. <b>Note:</b> DFHSMPE is installed in SDFHINST.
DFHSTART	Starts CICS. <b>Note:</b> DFHSTART is installed in SDFHINST.
DFHYBTPL	Translates, compiles, and link-edits PL/I application programs using EXEC DLI commands in a batch environment under Language Environment.
DFHYBTVL	Translates, compiles, and link-edits COBOL application programs using EXEC DLI commands in a batch environment under Language Environment.
DFHYITDL	Translates, compiles, and link-edits C/370™ application programs using the command-level interface under Language Environment.
DFHYITEL	Translates, compiles, and link-edits C++ application programs using the command-level interface under Language Environment.
DFHYITPL	Translates, compiles, and link-edits PL/I application programs using the command-level interface under Language Environment.
DFHYITVL	Translates, compiles, and link-edits VS COBOL application programs using the command-level interface under Language Environment.
DFHYXTDL	Translates, compiles, and link-edits C/370 application programs using the external CICS Transaction Server interface under Language Environment.
DFHYXTEL	Translates, compiles, and link-edits C++ application programs using the external CICS Transaction Server interface under Language Environment.
DFHYXTPL	Translates, compiles, and link-edits PL/I application programs using the external CICS Transaction Server interface under Language Environment.
DFHYXTVL	Translates, compiles, and link-edits VS COBOL application programs using the external CICS Transaction Server interface under Language Environment.
DFHZITCL	Translates, compiles, and link-edits COBOL application programs using the integrated CICS translator.

Table 7. CICS Transaction Server-supplied procedures (continued)

Procedure	Description
DFHZITDL	Translates, compiles, and link-edits XL C application programs using the integrated CICS translator.
DFHZITEL	Translates, compiles, and link-edits XL C++ application programs using the integrated CICS translator.
DFHZITFL	Translates, compiles, and link-edits XL C XPLINK application programs using the integrated CICS translator.
DFHZITGL	Translates, compiles, and link-edits XL C++ XPLINK application programs using the integrated CICS translator.
DFHZITPL	Translates, compiles, and link-edits PL/I application programs using the integrated CICS translator.
EYUCMASP	Starts a CMAS.
EYUCSYSP	Starts a MAS.
EYUWUIP	Starts a WUI.
ICCFCC	CICS foundation classes
ICCFCL	CICS foundation classes
ICCFGL	CICS foundation classes
ICCFCL	CICS foundation classes
<b>Note:</b>	
For further information about using the CICS Transaction Server-supplied procedures, see:	
<ul style="list-style-type: none"> <li>• “The CICS TS-supplied SMP/E procedure” on page 207, for information about DFHSMPE and DFHSTART</li> <li>• <code>../com.ibm.cics.ts.doc/dfha2/topics/overview.dita</code> in the <i>CICS System Definition Guide</i> for information about the other procedures.</li> </ul>	

The procedures DFHAUPLE, DFHSMPE, DFHSTART, EYUCMASP, EYUCSYSP and EYUWUIP are tailored to your CICS Transaction Server environment and stored in the CICSTS32.XDFHINST library when you run the DFHISTAR job. The other procedures are **not** modified by the DFHISTAR job and are copied into the CICSTS32.SDFHPROC library when you run the CICS Transaction Server installation jobs.

You should copy all these procedures into a cataloged procedure library (for example, SYS1.PROCLIB). **Before you copy the procedures**, read the following:

1. Your procedure library may already contain procedures, supplied with an earlier release of CICS, that have the same names as the new procedures but are, in fact, different. If so, you must find some way of selecting the right release. Here are some ways of using the new versions:
  - a. For the time being, rename either set of procedures, and modify the appropriate jobs to use the new names.
  - b. Insert the new procedures into the job streams that use them, and use the procedures as in-stream procedures. The inserted procedures should be placed between the JOB statement and the first EXEC statement. You must insert a `// PEND` statement after the inserted procedures. When the new release becomes the production system, you can copy the new procedures into your procedure library.
  - c. Indicate the DDNAME of the cataloged procedure library that is to be used to convert the JCL for the job. For example, you could use the JES2

/\*JOBPARM PROCLIB=xxxxxxx. For further information about specifying DDNAMEs in JCL, see the *z/OS MVS JCL Reference* manual.

2. If service is applied to the CICS Transaction Server procedures, it is the versions in the libraries CICSTS32.CICS.SDFHINST and CICSTS32.CICS.SDFHPROC that will be updated by SMP/E. You must then copy the updated procedures into your procedure library.
3. The default for the symbolic parameter GZONE in the procedures DFHSMPE and DFHAUPLE is taken from the value that you specified by the GZONE parameter of the DFHISTAR job.
4. The default for the ZNAME symbolic parameter in the procedures DFHSMPE and DFHAUPLE is taken from the value that you specified by the TZONE parameter of the DFHISTAR job. For a description of how the ZNAME parameter is used, see the SMP\_CNTL DD statement in The CICS TS-supplied SMP/E procedure.
5. Change the OUTC parameter as required.

When you have read these notes, and acted on them as necessary, copy the procedures into a cataloged procedure library.

## Create Extra Sets of CICS Transaction Server Target Libraries (Optional)

You can use the CICS Transaction Server installation job, DFHISTAR, to generate an optional installation job, DFHINSTA, which you can use to create extra copies of the CICS Transaction Server target libraries and UNIX System Services directories.

Some of the benefits of using multiple libraries are:

- **Backing out PTFs and APARs** - if you apply PTFs or APARs to CICS Transaction Server and if they fail a fix-test, you can back out the changes with minimum disruption.
- **DASD failure** - multiple libraries protect you against failure of the DASD on which the CICS Transaction Server load libraries reside.

Base the decision to use multiple libraries for CICS Transaction Server on the following factors:

- Your need for high availability - as already stated, the use of multiple libraries can protect you against CICS Transaction Server downtime due to DASD failure or incorrect service (either from IBM-supplied PTFs or your own modifications to your CICS Transaction Server region).
- The extra DASD needed - multiple libraries require more disk space.
- Other ways of providing high availability; for example, use of a CICSplex, VTAM persistent sessions, and MVS functions to provide restart of CICS Transaction Server regions.
- The added complexity of maintaining multiple sets of CICS Transaction Server libraries - two or more sets of CICS Transaction Server target libraries, together with the SMP/E procedures needed to support them, increase the complexity of maintenance. You will need to define procedures to ensure that upgrades to the CICS Transaction Server libraries are kept under control.
- Alternative solutions - if you have already established a proven process for fix verification and for testing applications developed for your production CICS Transaction Server region, you may decide you don't need multiple CICS Transaction Server libraries.

You can use the DFHINSTA job, generated by the DFHISTAR job, to create extra sets of CICS Transaction Server target libraries fully under the control of SMP/E. Each time you run the DFHINSTA job, you can only generate one extra set of target libraries.

To create an extra sets of target libraries, you should complete the following steps. You can repeat the steps to create more sets of target libraries.

1. Edit the DFHISTAR job to specify values for:

- The ADDTVOL, AINDEX, ASMPSCDS, AZONE, AZONECSI, AZONELOG, and USSDIRA parameters, for the new set of target libraries.
- The INDEX, TZONE, TZONECSI, and TZONELOG parameters, for the primary target libraries you want to copy from. (The TZONE, TZONECSI, and TZONELOG parameters must specify the target zone that contains the CICS Transaction Server target libraries defined with the high-level qualifier provided by the INDEX parameter.)
- The DZONE, DZONECSI, and DZONELOG parameters, for the distribution libraries to be associated with the new set of target libraries.
- The SELECT parameter, to specify DFHINSTA (that you want to copy) and the member name you want the generated version of DFHINSTA to be stored as in the CICSTS32.XDFHINST library. For example,

```
SELECT DFHINSTA INSTA111
```

will store the generated version of DFHINSTA into member INSTA111 of the CICSTS32.XDFHINST library when you submit the DFHISTAR job.

Each time you copy DFHINSTA (to create a new set of target libraries), you should specify a new name on the SELECT parameter (to save each copy with a different name in case you need it again in the future).

For further information about editing the DFHISTAR job, and about the parameters of the DFHISTAR job, see “Edit the DFHISTAR Job” on page 25. The other parameters in the DFHISTAR job should not be changed.

2. Submit the DFHISTAR job

When you run the DFHISTAR job, it saves the generated version of the DFHINSTA job in the CICSTS32.XDFHINST library with the member name specified on the SELECT parameter in the DFHISTAR job. The data set name of the CICSTS32.XDFHINST library is specified in the LIB parameter of the DFHISTAR job.

3. Consider running the DFHIFSA job

If you decide to create an additional SMP/E target zone using job DFHINSTA, run the DFHIFSA job first before running the DFHINSTA job. DFHIFSA creates an additional SMP/E target zone for z/OS UNIX. This job:

- Unmounts the z/OS UNIX file system at directory */pathprefix/usr/lpp/cicsts/ussdir* to allow the job to rerun, and if necessary forces return code 0.
- Deletes the *ussdira* directory at */pathprefix/usr/lpp/cicsts*, where *ussdira* is the name of the directory specified on the **ussdira** parameter in the DFHISTAR job. This allows the job to rerun, and if necessary forces return code 0.
- Deletes the file system specified in the HFSADSN parameter of the DFHISTAR job to allow the job to rerun, and if necessary forces return code 0.
- Creates the file system specified in the HFSADSN parameter of the DFHISTAR job

- Creates the */ussdira* directory at */pathprefix/usr/lpp/cicsts*, where *ussdira* is the name of the directory specified in the **ussdira** parameter in the DFHISTAR job
- Mount the file system at directory */pathprefix/usr/lpp/cicsts/ussdira*
- Changes the permission settings for the *ussdira* directory to 775

All steps of this job must end with return code zero for the job to be successful. CICS requires the MOUNT issued by DFHIFSA to access files stored in the z/OS UNIX, but the MOUNT command is lost when you re-IPL MVS. SDFHINST member DFHBPXA contains a MOUNT command for */pathprefix/usr/lpp/cicsts/ussdira*. Copy this command into a BPXPRMxx member of the SYS1.PARMLIB dataset to ensure the mount is restored when MVS is IPLed.

#### 4. Submit DFHINSTA

The DFHINSTA job (or a copy of it) copies the CICS Transaction Server target libraries specified by the INDEX parameter, and creates corresponding CICS Transaction Server SMP/E data sets for them. In particular, it allocates a new SMP/E CSI data set for the extra target zone.

So that DFHINSTA job can be run more than once, step 1 deletes previous copies of the data sets to be created. Step 3 deletes the SMP/E CSI data set. Step 6 removes the ZONEINDEX entry for the extra target zone.

The first time the DFHINSTA job is run, Step 6 will give the following messages:

```
GIM35701E ** ZINDEX SUBENTRY azone WAS NOT DELETED BECAUSE
                IT DOES NOT EXIST.
GIM25601I      THE SPECIFIED ENTRY WAS NOT UPDATED BECAUSE OF
                AN ERROR DURING UCLIN PROCESSING.
```

You can ignore these messages the first time the job is run.

## Load the CICS Transaction Server Source Material (Optional)

You can use the sample job, DFHOPSRC, to load the optional CICS Transaction Server source from the distribution tapes. The DFHOPSRC job is generated in the CICSTS32.XDFHINST library when you run the DFHISTAR job.

The DFHOPSRC job runs the MVS IEBCOPY utility program to load the optional source data sets from the tapes into a single DASD data set.

For further information about the IEBCOPY program, see the *MVS/ESA Data Administration: Utilities Manual* (SC26-4516).

**Note:** The DFHOPSRC job loads the tape files to a DASD device of your choice, using a BLKSIZE parameter of 6160. You can specify a different BLKSIZE parameter on the DCB and SPACE statements if you want, in which case the space allocation should be reviewed (It may need to be changed.).

## Load Other Optional Features

The sample job, DFHINSTJ, is supplied to be used to load other optional features that may be supplied. CICS Transaction Server V2R3 contains no optional features at General Availability time, but the sample job may be used if any optional features are shipped at a later date. The DFHINSTJ job is generated in the CICSTS32.XDFHINST library when you run the DFHISTAR job.

---

## Activating CICS Transaction Server

After you have loaded CICS Transaction Server to disk, you should perform the following steps:



1. (If needed) Apply service, see Chapter 29, “Applying service to CICS Transaction Server for z/OS,” on page 207
2. Integrate CICS Transaction Server with MVS
3. Create CICS Transaction Server system data sets, see “Creating the CICS data sets” on page 218
4. (Optional) Install CICS-IMS DBCTL support, see “Defining DL/I support” on page 230
5. (Optional) Install MRO and ISC support, see “Installing MRO and ISC support” on page 239
6. (Optional) Run the installation verification procedures (IVPs), see “Running the installation verification procedures” on page 323.

Information about how to get CICS Transaction Server into operational status, is also included as part of the process of verifying the installation, see “Running the installation verification procedures” on page 323.

---

## Checklist for the CICS Transaction Server for z/OS Installation

Table 8. CICS Transaction Server installation checklist

Step	Done?	Description	See page
1		Check that you have received the material that you ordered.	N/A
2		Check that you have all the installation prerequisites,	See the <i>Program Directory</i>
3		Copy RELFILE(2) from the distribution tape. (Record your name for the TDFHINST library: _____)	See the <i>Program Directory</i>
4		Edit the DFHISTAR job in the TDFHINST library	“Edit the DFHISTAR Job” on page 25
4.1		Specify the CICS Transaction Server temporary installation libraries	“Specify the CICS Transaction Server temporary installation libraries” on page 31
4.2		Specify the JOB parameters for installation jobs	“Specify the JOB parameters for installation jobs” on page 31
4.3		Specify the scope of the installation	“Specify the scope of the installation” on page 32

Table 8. CICS Transaction Server installation checklist (continued)

Step	Done?	Description	See page
4.4		Specify the type of JES to be used	“Specify the type of JES to be used” on page 32
4.5		Specify the utilities to be used	“Specify the utilities to be used” on page 32
4.6		Specify the prefix of CICS Transaction Server jobs	“Specify the prefix of CICS Transaction Server jobs” on page 33
4.8		Specify the prefix of the CSSLIB library	“Specify the Data Set Name of the CSSLIB Library” on page 46
4.9		Specify the prefix of the SISpload library	“Specify the Data Set Name of the SISpload Library” on page 46
4.10		Specify the indexes of CICS Transaction Server data sets	“Specify the indexes of CICS Transaction Server data sets” on page 33
4.11		Specify block sizes	“Specify block sizes” on page 35
4.13		Specify the disk unit for work data sets	“Specify the disk unit for work data sets” on page 35
4.14		Specify disk volumes	“Specify disk volumes” on page 36
4.15		Allocate the space for CICS Transaction Server disk volumes	“Allocate space for CICS Transaction Server disk volumes” on page 39

Table 8. CICS Transaction Server installation checklist (continued)

Step	Done?	Description	See page
4.16		Specify SMP/E zone attributes	“Specify SMP/E Zone Attributes” on page 41
4.17		Specify attributes of the temporary SMP/E work data sets	“Specify Attributes of the Temporary SMP/E Work Data Sets” on page 39
4.18		Specify attributes of the permanent SMP/E data sets	“Specify Attributes of the Permanent SMP/E Data Sets” on page 40
4.19		Specify the high-level qualifiers for SMP/E data sets	“Specify the High-level Qualifiers for SMP/E Data Sets” on page 44
4.20		Specify the z/OS UNIX directories and data sets	“Specify the CICS TS z/OS UNIX directories and data sets” on page 34
4.21		Specify the distribution tape device type	“Specify the distribution tape device type” on page 44
4.22		Specify attributes of the CICS Transaction Server system data sets	“Specify attributes of the CICS Transaction Server system data sets” on page 44
4.23		Specify attributes specific to CICSplex SM (optional)	“Specify attributes specific to CICSplex SM” on page 50

Table 8. CICS Transaction Server installation checklist (continued)

Step	Done?	Description	See page
4.24		Specify attributes of any additional target libraries	“Specify Attributes of any additional target Libraries” on page 45
5		Create RACF profiles for the CICS Transaction Server data sets	“Create RACF Profiles for the CICS Transaction Server Data Sets” on page 53
6		Run the DFHISTAR job	“Run the DFHISTAR Job” on page 53
6.1		Check the output from the DFHISTAR job	“Check the Output from the DFHISTAR Job” on page 54
7		Check that you are ready to run the installation jobs	“Check that You Are Ready to Run the Installation Jobs” on page 55
8		Run the installation jobs	“Run the Installation Jobs” on page 56
8.1		Check the output from the installation jobs	“Check the Output from the Installation Jobs” on page 61
9		Copy the CICS Transaction Server procedures into a procedure library	“Copy the CICS Transaction Server procedures into a procedure library” on page 62

Table 8. CICS Transaction Server installation checklist (continued)

Step	Done?	Description	See page
10		Create any extra sets of CICS Transaction Server target libraries (optional)	“Create Extra Sets of CICS Transaction Server Target Libraries (Optional)” on page 64
11		Load the CICS Transaction Server optional source material (optional)	“Load the CICS Transaction Server Source Material (Optional)” on page 66
12		Load the Japanese national language feature (optional)	“Load Other Optional Features” on page 66



---

## Chapter 6. Installing the CICS Information Center

The CICS Information Center runs in an Eclipse framework, commonly known as a *help system*, and consists of a number of documentation plug-ins, including various tools and connectors. The information center is supplied on CD-ROM with a readme file. The readme file contains a complete list of all the plug-ins that are provided, and you can select which plug-ins to install based on your CICS environment. If you already have an Eclipse help system, or an Eclipse-based IDE such as WebSphere Studio Enterprise Developer, you can opt to install only the CICS documentation plug-ins. You can run the information center locally on a workstation, or as a server with remote access. Before you begin to install the information center, read the following topics:

- “Requirements”
- “Installing the information center on a workstation”
- “Installing the information center on a server” on page 74
- “Installing the CICS plug-in in an Eclipse IDE or help system” on page 75

---

### Requirements

The information center is supported on the following platforms:

- Windows 2000
- Windows XP
- AIX® 5.2 and 5.3
- Linux RedHat Enterprise 3.0
- Linux SuSE Enterprise 3.0
- Linux RedHat Enterprise 8 and 9 for zSeries®
- Linux SuSE Enterprise 8 and 9 for zSeries
- z/OS 1.7 or later

Please note that support for the information center on Linux for zSeries and z/OS is only offered in server mode for remote access using a browser.

The information center uses a JRE. A JRE for each platform is provided with the information center, except for z/OS. If you want to run an information center on z/OS, you need to use the JRE provided with the operating system.

To get the best results when viewing the information center, it is recommended that you use one of the following browsers:

- Microsoft Internet Explorer 6.0
- Mozilla 1.7

To view PDF documents within the information center, you require Adobe Acrobat Reader 4.05 or higher installed, and the Acrobat Web plug-in installed in your browser.

---

### Installing the information center on a workstation

The information center can run locally from any directory on your workstation. If you do not have an Eclipse help system, you will need to install this first as it is required to run the information center. If you do have an Eclipse help system, see “Installing the CICS plug-in in an Eclipse IDE or help system” on page 75.

1. Select the appropriate folder for your operating system on the CD-ROM. This folder contains a zip file of the Eclipse help system.
2. Unzip the zip file to a new directory on your workstation. This creates an eclipse help system on your workstation.
3. Select the plug-ins folder on the CD-ROM. This folder contains all of the documentation plug-ins.
4. Copy the documentation plug-ins you require to the eclipse\plugins directory of the help system that you have just unzipped. The readme file contains a complete list of the plug-in names and the products they refer to. It is recommended that you copy the CICS TS plug-in `com.ibm.cics.ts.doc`, the CICS support plug-in `com.ibm.cicsts.doc`, and associated support plug-ins `com.ibm.support.core.doc` and `com.ibm.support.core.doc.nl`.
5. At the highest directory level in your Eclipse help system, execute the file `IC_local_Start` to start the information center on your workstation. This should automatically launch a browser and display the information center.

Please note that you cannot run the information center locally on z/OS 1.7 or the Linux for z/OS platforms.

---

## Installing the information center on a server

Running the information center on a server requires you to allocate a specific port number, to allow remote access using a browser. You can run the information center in server mode on any of the supported platforms listed in “Requirements” on page 73. If you already have a help system installed, see “Installing the CICS plug-in in an Eclipse IDE or help system” on page 75 for information on how to install just the CICS documentation plug-ins. To install the information center on UNIX, Linux and Windows servers, follow these steps:

1. Select the appropriate folder for your operating system on the CD-ROM. This folder contains a zip file of the Eclipse help system.
2. FTP the zip file to an appropriate directory on your server and then unzip it. This creates a help system on your server.
3. Select the plug-ins folder on the CD-ROM. This folder contains all of the documentation plug-ins.
4. Use the readme file on the CD-ROM to select which documentation plug-ins you want to install. Copy the required plug-ins to the eclipse\plugins directory of the help system that you unzipped. It is recommended that you copy the CICS TS plug-in `com.ibm.cics.ts.doc`, the CICS support plug-in `com.ibm.cicsts.doc`, and associated support plug-ins `com.ibm.support.core.doc` and `com.ibm.support.core.doc.nl`.
5. At the highest directory level in your Eclipse help system, edit the file `IC_server_start` to specify the port number that you want the information center to use. The default is 29127. If your company allows you to reserve ports, you can ask your TCP/IP system programmer to reserve this port or the port number of your choice.
6. Execute the file `IC_server_start` to start the information center in server mode.
7. To verify that you can access the information center, start up a browser and type the URL `http://servername:port`, where *servername* is the name of the server where you installed the information center and *port* is the port number that you specified in the start up file.

The information center runs in the UNIX System Services (USS) component of z/OS. To install the information center on z/OS, follow these steps:



1. Select the z/OS folder on the CD-ROM, which contains a tar file of the Eclipse help system.
2. FTP the tar file to a suitable z/OS UNIX directory in USS. Ensure you mount the z/OS UNIX file containing this directory through the BPXPRMxx member of SYS1.PARMLIB.
3. Un-tar the file using the command `tar -xvf filename`. This creates a help system on your server.
4. Select the plugins folder on the CD-ROM. This folder contains all of the documentation plug-ins.
5. Use the readme file on the CD-ROM to select which documentation plug-ins you want to install. FTP the required plug-ins to the `eclipse\plugins` directory of the help system that you unzipped. It is recommended that you copy the CICS TS plug-in `com.ibm.cics.ts.doc`, the CICS support plug-in `com.ibm.cicsts.doc`, and associated support plug-ins `com.ibm.support.core.doc` and `com.ibm.support.core.doc.nl`.
6. At the highest directory level in your Eclipse help system, edit the file `IC_server_start` to specify the directory path of a Java Runtime Environment (JRE) at 1.4.2 that you want the information center to use. You can also change the port number from the default of 29127. Use the command `vi IC_server_start.sh` to open the file to edit it, or if connected through TSO you can use the command `oedit IC_server_start.sh`. Add the following command to the beginning of the file, using the appropriate path to the JRE. For example, you could specify:

```
export PATH=/u/lpp/java142/J1.4/bin:$PATH
```
7. Execute the start file using the command `./IC_server_start.sh &`. The `&` indicates that the information center should run as a background task. This means that if you log off, the information center will continue to run on the server.
8. To verify that you can access the information center, start a browser and type the URL `http://servername:port`, where *servername* is the name of the server where you installed the information center and *port* is the port number that you specified in the start file.

---

## Installing the CICS plug-in in an Eclipse IDE or help system

If you already have an Eclipse help system installed on your workstation or server, or if you are using any Eclipse-based product (for example, all versions of WebSphere Studio Enterprise Developer or WebSphere Developer for zSeries), you can install the CICS documentation and run it within your existing software. Ensure that you shut down your help system or product before adding the CICS documentation plug-ins, otherwise Eclipse will not recognize the newly added plug-ins. Follow these steps:

1. Select the plugins folder on the CD-ROM. This folder contains all of the documentation plug-ins.
2. Use the readme file on the CD-ROM to select which documentation plug-ins you want to install. It is recommended that you copy the CICS TS plug-in `com.ibm.cics.ts.doc`, the CICS support plug-in `com.ibm.cicsts.doc`, and associated support plug-ins `com.ibm.support.core.doc` and `com.ibm.support.core.doc.nl`.
3. Copy the required documentation plug-ins to the `eclipse\plugins` directory of the help system or Eclipse Integrated Development Environment (IDE) that you already have installed. For example, for a default installation of WebSphere

Studio Enterprise Developer, you would copy the plug-ins to the directory  
C:\Program Files\IBM\WebSphere Studio\eclipse\plugins.

4. Start up the help system or Eclipse IDE to ensure that the CICS documentation is present.

---

## **Part 3. Setting up the MVS environment for CICS**

This part discusses what you should do after you have loaded the CICS Transaction Server elements to DASD, and before you run CICS.



---

## Chapter 7. Authorizing the CICS and CICSplex SM libraries

Ensure that the following libraries are APF-authorized:

- SDFHAUTH
- SEYUAUTH
- SDFHLINK
- SEYULINK
- SDFHLPA
- SEYULPA

If you require Java support, you should ensure that the SDFJAUTH library is APF-authorized. See “Authorizing the hlq.SDFJAUTH library” on page 238 for more information about that library.

Although, in general, CICS runs in problem state, the CICS initialization program, DFHSIP, needs to run in supervisor state for part of its execution. The CMAS startup program, EYU9XECS, also requires APF authorization.

For a module to be able to run in supervisor state, it must be link-edited as an authorized module into a partitioned data set, which must also be defined to the operating system as APF-authorized. For CICS-supplied modules, the link-editing has been done for you. The CICS-supplied DFHSIP module is link-edited with the **authorized** attribute (using SETCODE AC(1)), and is installed in the SDFHAUTH library.

APF-authorize these libraries by adding them to the list of APF-authorized libraries in the appropriate PROGxx (or IEAAPFxx) member in SYS1.PARMLIB. These libraries must be APF-authorized to enable certain modules, such as DFHSIP, to run in supervisor state.

If your list(s) of APF-authorized libraries are specified in the dynamic format (in a PROGxx member), refresh the APF list dynamically using the SETPROG or SETPROG=xx command.

If your list(s) of APF-authorized libraries are specified in the static format (in IEAAPFxx members), schedule an MVS IPL for the APF-authorization to take effect.

For information about maintaining lists of APF-authorized libraries, see the *z/OS MVS Initialization and Tuning Guide*.

When you prepare your startup job stream, provide a STEPLIB DD statement for the SDFHAUTH and SEYUAUTH libraries. When you define your STEPLIB DD statement, remember that all other libraries concatenated with the SDFHAUTH and SEYUAUTH libraries must also be APF-authorized. If any of the libraries in a STEPLIB concatenation are not authorized, MVS regards all of them as unauthorized.

The SDFHLOAD and SEYULOAD libraries contain only programs that run in problem state, and should **not** be authorized. The SDFHLOAD library must be included in the CICS DFHRPL library concatenation. There is an example of this library DD statement in the sample job stream provided in the *CICS System Definition Guide*.

| For information about authorizing access to CICS data sets, see the *CICS RACF*  
| *Security Guide*.

---

## Chapter 8. Authorizing CICS regions to access MVS resources

You should consider authorizing access to the following when planning your security requirements to run CICS:

### **CICS PDS libraries**

Protect your CICS data sets that use RACF. See “Protecting CICS load module data sets.”

### **VTAM ACB**

Authorize each CICS region userid to OPEN the VTAM ACB for the region's specified APPLID. See “Authorizing access to a CICS region's VTAM ACB” on page 82.

### **CICS system transactions**

Authorize each CICS region userid to access the CICS category 1 system transactions. See “Authorizing the region userid to access category 1 transactions” on page 83.

### **SMSVSAM server**

Authorize each CICS region to open the SMSVSAM control ACB if you plan to use CICS with VSAM record-level data sharing. See “Authorizing access to an SMSVSAM server” on page 83.

### **System logger log streams**

Authorize each CICS region userid to access the MVS system logger log streams that are used by CICS. See “Authorizing access to MVS log streams” on page 84.

### **UNIX System Services**

Include an OMVS segment in the CICS region's user profile, specifying the UID parameter, to ensure that your CICS regions have the required access to UNIX system services. See “Giving CICS regions access to z/OS UNIX System Services” on page 86.

### **RACF resource classes**

Activate the appropriate RACF resource classes to enable terminal users to access CICS resources and user-defined resources. See “Activating RACF resource classes” on page 89.

### **VSAM catalogs**

Grant the CICS region user ID read access to each VSAM catalog that contains files where CICS has the file definitions installed and where these files are opened during or after CICS startup. Include the VSAM catalog for the DFHCSD file for the CICS system definition data set (CSD).

---

## Protecting CICS load module data sets

To prevent unauthorized or accidental modification of *hlq.SDFHAUTH* or *hlq.SDFJAUTH*, you should RACF-protect these libraries. Without such protection, the integrity and security of your MVS system are at risk. Additionally, if you require protection against the unauthorized use of DFHSIP, do not place this module in the LPA and do not include *hlq.SDFHAUTH* in the MVS LNKLIST unless DFHSIP is RACF-protected as a controlled program with a profile in the RACF PROGRAM resource class.

You should also RACF-protect the other libraries (including *hlq.SDFHLOAD*) that make up the STEPLIB and DFHRPL library concatenations.

For information about authorizing access to CICS data sets, see the *CICS RACF Security Guide*.

## Authorizing access to data set services modules

During initialization, CICS determines the availability of backup-while-open (BWO) support by linking to the callable services modules IGWAMCS2 and IGWABWO. CICS also checks the DFSMSdss (or DFSS) release level by linking to the module ADRRELV. If access to this data set services module is controlled by means of RACF PROGRAM general resource profiles, security violation messages are issued against the CICS region userid, unless the userid is authorized to access ADR-prefixed module names.

You can avoid security violation messages against the CICS region userids, and still control access to data set services, as follows:

- If you have generic PROGRAM profiles protecting access to ADR modules, create specific PROGRAM profiles for the ADDRELV module, and ensure your CICS region userids have READ access to these specific profiles.
- Instead of using PROGRAM profiles to protect access to data set services, use one of the following methods:
  - Define suitable profiles in the DASDVOL general resource class.
  - Defining profiles in the FACILITY general resource class that are supported by DFSMS to control access to data set services.

For information about using DASDVOL and FACILITY class profiles to control the uses of data set services, see the *DFSMS/MVS DFSMSdss Storage Administration Reference*, SC26-4929, and the *DFSMS/MVS DFSMSdss Storage Administration Guide*, SC26-4930.

---

## Authorizing access to a CICS region's VTAM ACB

You can control which users, among those who are running non-APF-authorized programs, can OPEN the VTAM ACB associated with a CICS address space (CICS region). This ensures that only authorized CICS regions can present themselves as VTAM applications that provide services with this APPLID, thus preventing unauthorized users from impersonating real CICS regions. (Note that the CICS region userid needs the OPEN access, not the issuer of the SET VTAM OPEN command.)

To enable CICS to start up with external security, you must first have authorized the CICS region userid to open the CICS region's VTAM ACB with the applid specified on the APPLID system initialization parameter.

For each APPLID, create a VTAMAPPL profile, and give the CICS region userid READ access. For example:

```
DEFINE VTAMAPPL applid UACC(NONE) NOTIFY(userid)
PERMIT applid CLASS(VTAMAPPL) ID(cics_region_userid) ACCESS(READ)
```

The correct CICS APPLID to specify in the VTAMAPPL class is the specific APPLID, as specified in the CICS system initialization parameters. If you are using XRF (that is, if CICS is started with XRF=YES in effect), you must define two VTAMAPPL profiles — one each for both the active and alternate CICS region's specific APPLID (the second operand on the CICS APPLID startup option).

**Note:**



1. The VTAMAPPL class must be active and RACLISTed for this protection to be in effect; for example:
 

```
SETRPTS CLASSACT(VTAMAPPL) RACLIST(VTAMAPPL)
```
2. If a CICS region is not to use VTAM, you do not need to authorize the CICS region userid for the CICS applid.
3. If you do not control the opening of a CICS region's VTAM ACB, a new VTAM application started with the same applid as that of a running CICS region has the following effect:
  - The running CICS region performs a FORCECLOSE of its VTAM ACB and issues message DFHZC0101.
  - The running CICS region either terminates or continues, depending on your use of the XXRSTAT exit. (The default is to terminate.) If the CICS region continues, it no longer uses VTAM.
  - The new application opens the VTAM ACB with the specified applid.
  - If the first running CICS region used VTAM persistent sessions, the new application recovers any VTAM sessions that persist from that CICS region.

For information about creating VTAMAPPL profiles for CICS region applids, see the *CICS RACF Security Guide*. For information about the XXRSTAT exit, see the *CICS Customization Guide*.

---

## Authorizing the region userid to access category 1 transactions

To enable CICS to start up with external security, you must first have authorized the CICS region userid to access the category 1 system transactions. If the region userid does not have this authority at CICS startup, CICS issues message DFHXS1113, and terminates.

To give the region userid the authority to access the category 1 system transactions, edit and submit the sample job stream in Figure 3 to execute the CICS-supplied sample CLIST, DFH\$CAT1. This job uses the RACF commands in the CLIST to update the RACF database.

**Note:** Only a user with the RACF authority SPECIAL can execute the CLIST to update the RACF database.

```
//RACFMIG JOB 'accounting information',
//          CLASS=A,USER=userid,PASSWORD=password
//DEFINE EXEC PGM=IKJEFT01
//SYSPRINT DD SYSOUT=A
//SYSTSPRT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//SYSTSIN DD *
EXEC 'CICSTS32.CICS.SDFHSAMP(DFH$CAT1)' LIST
/*
//
```

Figure 3. Batch job to execute the sample CLIST, DFH\$CAT1

For more information, see the *CICS RACF Security Guide*.

---

## Authorizing access to an SMSVSAM server

If you plan to run CICS with VSAM record-level sharing (RLS), you must authorize each region that connects to an SMSVSAM server to have access to that server.

In a test environment you might wish to use the default action and allow any CICS region using VSAM RLS to connect to an SMSVSAM server. If you wish to protect this access, the RACF SUBSYSNM general resource class must be active and you must authorize each CICS region that connects to an SMSVSAM server to have access to that server. This means granting access to the appropriate profile in the RACF SUBSYSNM general resource class. You define profiles in the SUBSYSNM resource class to control access by subsystems like CICS that want to connect to SMSVSAM.

A SUBSYSNM profile name is the name by which a given subsystem, such as CICS, is known to VSAM. For CICS regions, you must use the CICS applid as the profile name in the SUBSYSNM general resource class.

When CICS attempts to register the control ACB during CICS initialization, SMSVSAM calls RACF to check that the CICS region userid is authorized to a profile name in the SUBSYSNM class that matches the CICS applid. If the CICS region userid does not have READ authority, the register fails.

For example, if the applid of a CICS AOR is CICSDA1, and the CICS region userid (shared by a number of AORs) is CICSDA# # , define and authorize the profile as follows:

```
RDEFINE SUBSYSNM CICSDA1 UACC(NONE) NOTIFY(userid)
PERMIT CICSDA1 CLASS(SUBSYSNM) ID(CICSDA# # ) ACCESS(READ)
```

You can use wildcard characters on the applid to specify more than one CICS region, for example:

```
PERMIT CICSDD%% CLASS(SUBSYSNM) ID(CICSDDGRP) ACCESS(READ)
```

---

## Authorizing access to MVS log streams

There is no facility within CICS for controlling LOGSTRM security checks. This is controlled by the MVS security administrator activating the LOGSTRM and FACILITY general resource classes by means of the SETROPTS command.

Users of the IXCMIAPU administrative data utility and CICS regions both require appropriate authorizations to log streams and IXLSTR coupling facility structures.

## Authorizations for users of IXCMIAPU

You create log structures and define log streams using the IXCMIAPU administrative data utility to update the LOGR data set. To do this, your userid needs the appropriate level of authorization, as shown in the following examples:

### Coupling facility structures

To define and delete log structures using IXCMIAPU, you need ALTER access to the LOGR resource profile named MVSADMIN.LOGR in the FACILITY general resource class. For example, use the following RACF command:

```
PERMIT MVSADMIN.LOGR CLASS(FACILITY) ACCESS(ALTER) ID(your_userid)
```

### Coupling facility log streams

To define, delete, and update log streams (including log stream models) that are defined in coupling facility structures, you need:

- ALTER access to the appropriate log stream profile defined in the LOGSTRM general resource class

- UPDATE access to the coupling facility structure (IXLSTR) profile defined in the FACILITY general resource class (in this case, profile names are prefixed with IXLSTR).

For example, if the log stream and structure resource profiles are defined to RACF with the following commands:

```
RDEFINE LOGSTRM log_stream_profile UACC(NONE) [NOTIFY]
RDEFINE FACILITY IXLSTR.structure_name_a UACC(NONE) [NOTIFY]
```

use the following RACF commands to give your userid the required authorizations to these two profiles:

```
PERMIT log_stream_profile CLASS(LOGSTRM) ACCESS(ALTER) ID(your_userid)
PERMIT IXLSTR.structure_name_a CLASS(FACILITY) ACCESS(UPDATE) ID(your_userid)
```

## Authorizations for CICS regions

If the LOGSTRM resource class is active, the level of authorization required depends on whether log streams are always explicitly defined to the MVS system logger.

Ensure that the CICS region userid is authorized to write to (and create if necessary) the log streams and log stream datasets that are used for its system log and general logs (see Chapter 27, “Defining the logger environment for CICS journaling,” on page 169.) You do this by granting the appropriate access authorization to log stream profiles in the RACF LOGSTRM general resource class:

- If CICS is expected to create log streams dynamically, CICS must have **ALTER** authority to the relevant log stream (LOGSTRM) profiles, and **UPDATE** authority to the relevant coupling facility structure (IXLSTR and IXGLOGR) profiles. For example:

```
PERMIT region_userid.applid.* CLASS(LOGSTRM) ACCESS(ALTER)
      ID(region_userid)
PERMIT IXLSTR.structurename CLASS(FACILITY) ACCESS(UPDATE)
      ID(region_userid)
PERMIT IXGLOGR.region_userid.* CLASS(DATASET) ACCESS(UPDATE)
      ID(region_userid)
```

- If all the log streams that CICS writes to are already defined to MVS, CICS needs only **UPDATE** authority to the log stream profiles. For example:

```
PERMIT region_userid.applid* CLASS(LOGSTRM) ACCESS(UPDATE)
      ID(region_userid)
PERMIT IXGLOGR.region_userid.* CLASS(DATASET) ACCESS(UPDATE)
      ID(region_userid)
```

**Note:** In the above examples, *region\_userid.applid.\** is the generic profile name of the log stream resource. These examples illustrate a resource name prefixed by the region userid and applid. *region\_userid* is the CICS region userid under which CICS is running, either as a started task or batch job.

Permit READ access to those users who need to read the CICS log streams. You must permit UPDATE access to those users who need to update journals by granting the user the appropriate authority to the log stream (in the LOGSTRM resource class) and to the JOURNALNAME (in the JCICSJCT class). You also need to grant access to the dataset profile protecting the dataset that contains the log stream.

The userid needed for the JCICSJCT is the logon userid, not the CICS region userid. The following example shows how to define this:

```
PERMIT journal_name CLASS(JCICSJCT) ACCESS(UPDATE) ID(logon_userid)
```

The generic profile in the following example could be defined to cover all the log streams referenced by the CICS region identified by its region userid and applid:

```
RDEFINE LOGSTRM region_userid.** UACC(NONE)
```

If, however, you have multiple CICS systems sharing the same region userid, but with differing security requirements, include the applid in the generic profile, as follows:

```
RDEFINE LOGSTRM region_userid.applid.* UACC(NONE)
```

The following example allows the CICS region userid under which CICS is running to write journal and log records to log streams in the named coupling facility structure:

```
PERMIT IXLSTR.structurename CLASS(FACILITY) ACCESS(UPDATE)  
ID(region_userid)
```

The following examples give access to two categories of user:

```
PERMIT region_userid.applid.* CLASS(LOGSTRM) ACCESS(READ)  
ID(authorized_browsers)  
PERMIT region_userid.applid* CLASS(LOGSTRM) ACCESS(UPDATE)  
ID(archive_userid)
```

In these examples, *archive\_userid* is the userid under which an application program runs to purge old data from CICS logs when the data is no longer needed, and *authorized\_browsers* refers to the userids of users allowed to read log streams, but cannot purge data.

If several CICS regions share the same CICS region userid, you can make profiles more generic by specifying \* for the *applid* qualifier.

The number of profiles you define depends on the naming conventions of the logs, and to what extent you can use generic profiling.

---

## Giving CICS regions access to z/OS UNIX System Services

CICS requires access to z/OS UNIX System Services, and to directories and files in the file system. CICS facilities that require this access include TCP/IP support, Java support, CICS Web support, and task-related user exits that are enabled with the OPENAPI option (including the task-related user exit of the CICS-DB2 attachment facility).

When a CICS region requests a z/OS UNIX function for the first time, RACF:

- Verifies that the user (the CICS region userid) is defined as a z/OS UNIX user.
- Verifies that the user's current connect group is defined as a z/OS UNIX group.
- Initializes the control blocks needed for subsequent security checks.

Follow the steps listed in this topic to ensure that each CICS region meets the security requirements. You will need to:

1. Choose a RACF group that all your CICS regions can use to access z/OS UNIX, and give a z/OS UNIX group identifier (GID) to this RACF group.
2. Give a z/OS UNIX user identifier (UID) to each CICS region userid.
3. Make sure that each CICS region userid connects to the RACF group that you chose.
4. Set up a home directory on z/OS UNIX for each of your CICS regions.

The UID and GID are numbers that can be in the range 0 to 16 777 216. (0 is a superuser ID.) Give some thought to naming conventions, and to any existing UIDs and GIDs in your z/OS UNIX system. *z/OS UNIX System Services Planning*, GA22-7800, explains how to manage the UIDs and GIDs for your z/OS UNIX system.

To assign a z/OS UNIX UID and GID for your CICS regions and set up a home directory:

1. Choose a RACF group that can be used by all your CICS regions. For example, you could use a RACF group that is defined as the default group of your CICS region user IDs, or you could set up a RACF group to be used only for access to z/OS UNIX System Services. When you are setting up facilities such as Java support or CICS Web support, you might want to use this RACF group for giving file access permissions on z/OS UNIX, in which case the RACF group's z/OS UNIX group identifier (GID) will be associated with these directories and files. This means that the owner of these directories and files, and anyone who is not the owner but needs to carry out operations with these files, will need to have this group as his or her group or one of their supplementary groups. The *CICS RACF Security Guide* explains how RACF groups work.
2. Choose a suitable z/OS UNIX group identifier (GID) for the RACF group, and assign the GID to the RACF group. To assign a GID, specify the GID value in the OMVS segment of the RACF group profile. For example, if the RACF group is CICSTSAB, and the GID you want to assign is 9, use the command:  

```
ALTGROUP CICSTSAB OMVS(GID(9))
```
3. Choose a suitable z/OS UNIX user identifier (UID) for each CICS region. The *CICS RACF Security Guide* explains how the region userid under which CICS executes is specified when CICS is run as a started task, as a started job, or as a job.
  - a. Assign your chosen UIDs to each of your CICS region userids. To assign UIDs, specify the UID value in the OMVS segment of the RACF user profile for each CICS region userid.
  - b. Also specify the name of a home directory for each CICS region using the HOME option. The directory name should be in the format `/u/CICS region userid`.

The *CICS RACF Security Guide* tells you how to update a RACF user profile using the ALTUSER command. For example, if the CICS region userid is CICSHAA1, and the UID you want to assign is 2001, use the command:

```
ALTUSER CICSHAA1 OMVS(UID(2001) HOME('/u/cicshaa1'))
```

If you want to know about the other information that can be specified in an OMVS segment parameter in a user profile besides the UID and home directory, see the *z/OS Security Server RACF Command Language Reference*, SA22-7687.

**Note:** It is possible to assign the same UID to more than one CICS region userid. If all your CICS regions need to use the same z/OS UNIX files (for example, the files required for Java support), you could give all the CICS regions the same UID, and then you could use that UID for giving file access permissions on z/OS UNIX. However, bear in mind that:

- a. The sharing of UIDs allows each CICS region to access all of the z/OS UNIX resources that the other CICS regions with that shared UID can access, and this might not be appropriate in your system.

- b. The sharing of UIDs is not normally recommended in a z/OS UNIX system.
  - c. If you do choose to share UIDs, note that the z/OS UNIX System Services parameter MAXPROCUSER limits the maximum number of processes that a single user (that is, with the same UID) can have concurrently active. *z/OS UNIX System Services Planning*, GA22-7800, has more information about this parameter.
4. Set up each of the z/OS UNIX directories that you have specified as a home directory for one of your CICS regions.
- a. If you are not using an automount facility, use the `mkdir` command to create the z/OS UNIX directories. For example, issuing the UNIX command
 

```
mkdir /u/cicshaa1
```

 creates the z/OS UNIX directory `/u/cicshaa1`. (If you are using the TSO command, the directory name must be enclosed in single quotes.)
  - b. Whether or not you are using an automount facility, allocate a z/OS UNIX data set for each directory. *z/OS UNIX System Services Planning*, GA22-7800, tells you how to do this.
  - c. If you are not using an automount facility, mount the data set that you have allocated. Again, *z/OS UNIX System Services Planning*, GA22-7800, tells you how to do this.

Note that the z/OS UNIX data set that you allocate for a CICS region's home directory has a finite size, and if a particular CICS region is using the home directory extensively, you might need to increase the amount of space that the region has available.

5. Verify that the permission defined for the `/etc` directory is set to 755, so that CICS can access the files. The `/etc` directory provides a symbolic link to the `/SYSTEM/etc` directory. The `/SYSTEM/etc` directory is created with a permission of 700, so you need to check that the `/SYSTEM/etc` directory permission is set to 755.
- a. Check the permission set in the `/SYSTEM/etc` directory, from the Unix shell:
 

```
ls -ld /SYSTEM/etc
```
  - b. If the permission is not `drwxr-xr-x`, issue the following Unix shell command to set permission to 755:
 

```
chmod 755 /SYSTEM/etc
```

If permission is not set to 755, you might receive a RACF error indicating that you have insufficient authority for CICS to access the files.
6. Make sure that each of your CICS region userids connects to the RACF group to which you assigned a z/OS UNIX group identifier (GID). If your CICS region userids need to connect to more than one RACF group, RACF list of groups must be active in your system.

Your CICS regions now have access to z/OS UNIX System Services. When you set up facilities such as Java support or CICS Web support, you will use the UIDs or GIDs to give the CICS regions permission to access directories and files on z/OS UNIX.

To check the UID and GID details for a user, use the `id` command in the UNIX environment. For example, issuing the `id` command for our example CICS region userid `CICSHAA1` would give the following result:

```
uid=2001(CICSHAA1) gid=9(CICSTSAB)
```

If you need more general information about RACF facilities for controlling access to z/OS UNIX System Services, see the *z/OS Security Server RACF Security Administrator's Guide*, SA22-7683.

---

## Activating RACF resource classes

Before you can use RACF for CICS resources and for user-defined resources, you must activate the associated RACF resource classes by using the RACF SETROPTS command.

To run the CICS-supplied IVPs with external security, you must activate the resource classes for CICS resources.

To use your own user-defined resources with external security in your CICS environment, you must:

- Define resource classes for your resources.
- Activate the resource classes.
- Optionally RACLIST resource classes to be used for QUERY SECURITY commands. This builds in-storage profiles for those resource classes.

For information about RACF resource classes, see the *CICS RACF Security Guide*.





---

## Chapter 9. Defining the default CICS userid to RACF

If you intend to use RACF to authorize terminal users to access CICS resources, you should define a default CICS userid to RACF and specify it on the CICS system initialization parameter, DFLTUSER. This default userid assigns the security attributes to be used for all CICS terminal users who do not sign on with the CESN transaction (or a user-written equivalent).

During startup, CICS tries to sign on the default userid. If it is not signed on (for example, if not defined), CICS issues message DFHSN0701 and terminates CICS initialization. After the valid default CICS userid is signed on, its security attributes are used for all CICS terminal users who do not sign on with the CESN transaction. If the default userid is defined to RACF with a CICS segment, the operator attributes in that segment are also used for users who do not sign on.

For information about defining the userid to RACF, see the *CICS RACF Security Guide*.



---

## Chapter 10. Installing CICS-required modules in the MVS linklist

There are two categories of modules that CICS TS loads from the MVS linklist:

1. CICS- and CICSplex SM-supplied modules
2. Modules of other MVS products (for example, DFSMS)

---

### CICS- and CICSplex SM-supplied modules required in the MVS linklist

CICS TS supplies the modules listed in the SDFHLINK, library or SEYULINK library (for the CICSplex SM modules), when you install CICS TS.

The CICS modules in SDFHLINK are:

**AMDUSREF**

Alias of DFHTG650.

**AXMSC**

AXM server connection routines for CICS data sharing servers.

**AXMSI**

AXM subsystem initialization routine for CICS data sharing servers.

**DFHDTCV**

Connection validation subroutine for shared data tables.

**DFHDTSVC**

Shared data tables SVC services.

**DFHGTCNV**

Subroutine used by LOGR subsystem interface.

**DFHLGCNV**

Exit routine for LOGR subsystem interface.

**DFHMVRMS**

General MVS RESMGR exit stub.

**DFHNCIF**

Named counter server interface.

**DFHNCOPT**

Named counter server options.

**DFHPD650**

Dump formatting routine for use with IPCS.

**DFHRPDUF**

System dump formatting routine for ONC RPC.

**DFHRPTRI**

Trace interpretation routine for ONC RPC.

**DFHRXSVC**

RRS domain authorized services.

**DFHSNNFY**

RACF CICS segment changes notification routine.

**DFHSNPTO**

CICS RACF dynamic parse TIMEOUT keyword print routine.

| **DFHSNVCL**

| CICS RACF dynamic parse OPCLASS validation routine.

| **DFHSNVID**

| CICS RACF dynamic parse OPIDENT validation routine.

| **DFHSNVPR**

| CICS RACF dynamic parse OPPTRY validation routine.

| **DFHSNVTO**

| CICS RACF dynamic parse TIMEOUT validation routine.

| **DFHSSIN**

| CICS subsystem that initializes the console message handling facilities.

| **DFHSSMGT**

| CICS subsystem message table that contains the text of messages for the  
| subsystem interface modules.

| **DFHTG650**

| Link module for the CICS GTF trace printing load module DFHTRGTF.

| **DFHTR650**

| Link module for the CICS GTF trace printing load module DFHTR650.

| **DFHTT650**

| Link module used for trace interpretation.

| **Note:** It is not essential that DFHTT650 be included in the MVS Linklist,  
| although it is recommended for the reasons given at the end of this  
| list. If it is not in the linklist, an APF authorized library in the  
| STEPLIB concatenation of the batch job is needed for every job  
| which needs the module.

| **DFHXCSVC**

| External CICS interface (EXCI) SVC services routine.

| The CICSplex SM modules in SEYULINK are:

| **EYU9D320**

| Dump formatting routine for use with IPCS.

| **EYU9X320**

| In each MVS image that contains a CMAS. EYU9X320 is the initialization  
| module for the ESSS.

| **EYU9A320**

| In each MVS image that contains a CMAS where you wish to run the  
| CICSplex SM API. EYU9A320 is the CICSplex SM API subtask module.

| These modules are supplied in an APF-authorized library in the MVS linklist  
| because:

- | 1. They can be required by non-CICS regions such as batch jobs or a CICS data  
| sharing server.
- | 2. They must be consistent across several CICS regions.
- | 3. They can be required by both CICS and non-CICS regions.
- | 4. The RACF dynamic parse routines are required by the Security Administrator  
| who executes the ADDUSER or ALTUSER commands under TSO. For  
| information about the RACF interface routines, see the *CICS RACF Security  
| Guide*.

Ensure the modules supplied in SDFHLINK and SEYULINK are available from an APF-authorized library in the MVS linklist by:

- Adding these modules, as required, to an existing APF-authorized library defined in the MVS linklist, or
- Defining SDFHLINK and SEYULINK as APF-authorized libraries and including them in the MVS linklist.

## Compatibility with earlier CICS releases

Unless otherwise stated, the CICS Transaction Server for z/OS, Version 3 Release 2 levels of the modules in SDFHLINK are compatible with earlier releases of CICS. The CICSplex SM modules in SEYULINK are not compatible with earlier releases. CICSplex SM modules in SEYULINK are release specific. If you plan to run multiple releases of CICSplex SM on the same MVS image, you must have the equivalent modules specific to the releases you are running.

**Note:** DFHPD650, DFHTG650, DFHTR650, and DFHTT650 are release dependant. If you run more than one release of CICS, ensure the correct versions are available (for example, DFHPD630 for CICS TS for z/OS, Version 2.3).

## CICS shared data tables modules for the MVS linklist

CICS supplies the following modules, for the shared data tables facility, in the *hlq.SDFHLINK*. If you intend using the shared data tables facility, ensure that these modules are available in the MVS linklist or the MVS link pack area:

- DFHDTSVC and DFHDTCV, because all regions using shared data tables must use the same level of SVC code
- DFHMVRMS, the RESMGR exit stub, because CICS JOBLIB/STEPLIB data sets are unavailable at end-of-memory

---

## CICSplex SM API linklist modules

In each MVS image that contains a CMAS where you plan to run the CICSplex SM API, ensure that the following required modules reside in the proper location. These modules are supplied in the SYS1.CICSTS32.CPSM.SEYUAUTH library.

### EYU9AB00

In an authorized library in either the MVS linklist or the STEPLIB concatenation of the application that calls the API. EYU9AB00 is the API batch interface module.

### EYU9XESV

In an authorized library in either the MVS linklist or the CMAS STEPLIB concatenation. EYU9XESV is the API security exit module.

In addition, any application that calls the API must be link-edited with one of the following stub routine modules, regardless of what programming language is used:

### EYU9ABSI

For batch, TSO, or NetView® programs. EYU9ABSI is supplied in the SYS1.CICSTS32.CPSM.SEYUAUTH library.

### EYU9AMSI

For application programs running in CICS. EYU9AMSI is supplied in the SYS1.CICSTS32.CPSM.SEYULOAD library.

## Installing the REXX function package

The REXX run-time interface to the API is supplied as a function package and host command environment. The interface consists of a single load module that contains two entry points:

### **EYU9AR00**

The function package

### **EYU9AR01**

The host command

EYU9AR00 is supplied in the SYS1.CICSTS32.CPSM.SEYUAUTH library with an alias of IRXFLOC.

For a REXX program to access the function package, the module EYU9AR00, along with its alternate entry point, EYU9AR01, and its alias, IRXFLOC, must reside in an authorized library in one of these places:

- The MVS linklist
- The STEPLIB concatenation of the application that calls the API.

For a REXX program to access the function package from NetView, the EYU9AR00 module must also be aliased to DSIRXLFP and placed in an authorized library in either the MVS linklist or the STEPLIB concatenation for the NetView system.

**Note:** Users of the CICSplex SM run-time interface are subject to the normal CICSplex SM API security checks. See the information in the *CICS RACF Security Guide*.

The following members contain SMP/E user modification control statements that you can use to move the necessary API load modules to the SYS1.CICSTS32.CPSM.SEYULINK library. These members are supplied in CICSTS32.CPSM.SEYUSAMP.

<b>Member</b>	<b>Load module</b>
EYU\$UM11	EYU9AR00
EYU\$UM12	EYU9AB00
EYU\$UM13	EYU9XESV

If you use the IRXFLOC or DSIRXLFP aliases to provide access to the REXX function package, they must be placed ahead of any other IRXFLOC or DSIRXLFP modules in the STEPLIB (or MVS linklist) concatenation.

If you do not want to use the aliases for the REXX function package, you must modify your REXX parameter modules (IRXPARMS, IRXTSPRM, and IRXISPRM). If you do this, the following is recommended:

- The function package supplied by CICSplex SM should be added as a System function package, rather than a Local or User function package.
- A new host command entry like the following should be added:
  - An 8-byte Command Environment name of 'CPSM
  - An 8-byte Command Routine name of 'EYU9AR01'
  - A 16-byte Command Token of blanks

As the last step in installing the REXX function package, you must:

- Increase the number of entries in the appropriate function package table.
- Add an entry to that table for EYU9AR00.

For more information about REXX function packages and host commands, see the *TSO/E Version 2 REXX/MVS Reference* book.

---

## Modules of other MVS products in the MVS linklist

There are some DFSMS modules that CICS loads from the MVS linklist. This requirement is either dependent on the function you are using (such as backup-while-open (BWO) support), or on the release of DFSMS. The modules are:

### **IGWABWO**

This module, supplied in the MVS callable services library, SYS1.CSSLIB, is loaded by CICS from the MVS linklist if you are using BWO for files accessed in non-RLS mode.

**Note:** In addition to IGWABWO being in the linklist, IGWAMCS2 must be installed in the LPA. CICS tests for the presence of this module in the LPA to determine that BWO support is present in the MVS image before attempting to load IGWABWO.

For files that are accessed in RLS mode, CICS does not require IGWABWO or IGWAMCS2.

### **IGWARLS**

CICS loads this module, supplied in the MVS callable services library SYS1.CSSLIB, from the MVS linklist. CICS issues the following message if it can not load IGWARLS:

```
DFHFC0116 APPLID THE LOAD OF CALLABLE
          SERVICE IGWARLS HAS FAILED WITH RETURN CODE
          X'EEEE'.
```

CICS initialization fails if CICS cannot load this callable services module.





---

## Chapter 11. Defining CICS as an MVS subsystem

CICS must be defined as an MVS subsystem before you can use any of following facilities:

- The console message-handling facility
- Multiregion operation (MRO)
- CICS shared data tables
- External CICS interface (EXCI).
- Extended restart facility (XRF)

Also, if you are running CICS with XRF in a multi-MVS environment or a two-CPC (central processing complex) configuration, defining CICS as an MVS subsystem can reduce operator intervention during takeover if MVS or a CPC fails while more than one CICS is running.

For information about the console message-handling facility see the *CICS Operations and Utilities Guide*.

For information about MRO, see the *CICS Intercommunication Guide*.

Defining CICS as an MVS subsystem involves three members of the SYS1.PARMLIB partitioned data set: IEASYSxx, IEFSSNaa, and DFHSSlly. You only need member DFHSSlly if you want the console message-handling facility.

**Note:** aa,xx,yy represent suffixes that are used to distinguish different versions of members of the SYS1.PARMLIB library.

Note that if you intend to start CICS with the START command you must either:

- Give the MVS started task procedure a name different from the subsystem name in IEFSSNaa (default 'CICS'), or
- Issue the start command with the parameter SUB=JES2 or SUB=JES3 as appropriate.

For more information about the subsystem interface, see the *z/OS MVS Using the Subsystem Interface* manual.

The following topics cover:

- “The IEASYSxx MVS initialization member”
- “Noting IEASYSxx values for CICSplex SM” on page 100
- “Updating IEASYSxx (CMAS)” on page 101
- “The IEFSSNaa MVS subsystem initialization member” on page 101
- “The SYS1.PARMLIB(BPXPRMxx) parameters” on page 102
- “The console message-handling facility” on page 103
- “EXCI pipe allocation” on page 108

---

### The IEASYSxx MVS initialization member

In an IEASYSxx member (of the SYS1.PARMLIB library) used for MVS initialization, include the parameter SSN=aa, where aa refers to the SYS1.PARMLIB member IEFSSNaa that contains the definitions for all subsystems needed for this IPL of MVS, including the definition of CICS as an MVS subsystem.

---

## Noting IEASYSxx values for CICSplex SM

Some of the MVS initialization values located in an IEASYSxx member of the SYS1.PARMLIB library are referenced during installation of CICSplex SM address spaces.

Access the IEASYSxx member of the SYS1.PARMLIB library used to initialize your MVS system and make note of the values assigned to the following parameters:

- APF=** Completes the name of the parmlib member (IEAAPFxx) that contains authorized library names.
- CMD=** Completes the name of the parmlib member (COMMNDxx) that contains commands to be issued internally during master scheduler initialization.
- LNK=** Completes the name of one or more parmlib members (LNKLSTxx) that contain names of data sets that are to be concatenated to SYS1.LINKLIB.
- LNKAUTH=**  
Specifies whether all data sets in the LNKLST concatenation are to be treated as APF authorized or whether only those that are named in the APF table are to be treated as APF authorized.
- LPA=** Completes the name of one or more parmlib members (LPALSTxx) that are concatenated to SYS1.LPALIB for the purpose of building the pageable LPA (PLPA and extended PLPA).
- MAXCAD=**  
Specifies the maximum number of SCOPE=COMMON data spaces to be allowed during an IPL.
- MAXUSER=**  
Specifies a value that the system uses (along with the RSVSTRT and RSVNONR parameter values) to limit the number of jobs and started tasks that the system can run concurrently during a given IPL.
- NSYSLX=**  
Specifies the number of linkage indexes (LXs), in addition to those in the system function table, to be reserved for system linkage indexes (LXs).
- PROG=**  
Completes the name of the parmlib member (PROGxx) that contains authorized library names when a dynamic APF list is being used.
- RSVNONR=**  
Specifies the number of address space vector table (ASVT) entries to be reserved for replacing those entries marked nonreusable for the duration of an IPL.
- RSVSTRT=**  
Specifies the number of ASVT entries to be reserved for address spaces created in response to a START command.
- SYSNAME=**  
Specifies the name of the system being initialized.

For more information about these parameters, see the *z/OS Initialization and Tuning Reference* manual.

---

## Updating IEASYSxx (CMAS)

In every z/OS image that contains a CMAS, you need to verify that the IEASYSxx member of the SYS1.PARMLIB library that you use for z/OS initialization includes the parameters:

### **MAXCAD=nnn**

Set or increase the value to include the number of common MVS data spaces that are needed for each CMAS. Each CMAS needs a minimum of 6 common MVS data spaces. When setting the MAXCAD limit, allow for 6 common MVS data spaces per CMAS, in addition to any common data spaces that may be in use by other products.

### **NSYSLX=nnn**

Set or increase the value to include the minimum number of linkage indexes (LXs) that are required by CICSplex SM. One LX is needed for the Environment Services System Services (ESSS), so the minimum number of LXs required for use by CICSplex SM is 1.

For additional information about these parameters, see the *z/OS Initialization and Tuning Reference* manual.

---

## The IEFSSNaa MVS subsystem initialization member

To define CICS as an MVS subsystem, code an entry in the IEFSSNaa member in the SYS1.PARMLIB library. If you want to use the console message handling facility, or to change the number of pipes that can be allocated in an EXCI address space, code the entry by using one of the following methods:

```
CICS,DFHSSIN,DFHSSIyy
```

or

```
SUBSYS SUBNAME(CICS)  
  INITRTN(DFHSSIN)  
  INITPARM(DFHSSIyy)
```

This entry is used for every CICS region, that you have IPLed with this version of the IEFSSN member, that runs under MVS. You do not have to specify both DFHSSIN and DFHSSIyy, however apart from the suffix yy, you must code the entry for each parameter using the exact format given in the example. The meanings of the terms are as follows:

**CICS** is the name of the CICS subsystem.

### **DFHSSIN**

is the name of the CICS subsystem routine that initializes the console message-handling facilities, and the number of pipes that can be allocated in an EXCI address space. If you omit this name, CICS is defined as an MVS subsystem, but none of the console message-handling facilities are enabled, and the default number of pipes, that can be allocated in an EXCI address space will be used. That default is 100..

### **DFHSSIyy**

is the name of a SYS1.PARMLIB member, described below, in which you have defined initialization parameters for message-formatting and EXCI pipe allocation for the CICS subsystem. If you specify DFHSSIN but omit DFHSSIyy, the DFHSSIN routine tries to use the parameters that are defined in member DFHSSI00.

If the DFHSSI00 member does not exist, the routine uses the default values.

- For message formatting these are defined in the DFHSSIN member. They are described in “Default message-formatting initialization parameters” on page 105.
- For EXCI pipe allocation, this is the fixed default value, 100.

The IEFSSNaa member in the SYS1.PARMLIB library also contains the definitions for all the other subsystems needed for this IPL of MVS, for example JES2, IRLM, and DATABASE 2 (DB2).

---

## The SYS1.PARMLIB(BPXPRMxx) parameters

Use of certain CICS functions and running a large number of CICS systems in an LPAR mean that the default options, shipped in the BPXPRMxx members of SYS1.PARMLIB, are not sufficient.

You must review the default options for the following functions:

- C and C++ programs compiled using the XPLINK compiler option
- Programs that run on open TCBs and use APIs other than the CICS API
- SSL TCBs specified by the MAXSSLTCBS system initialization parameters
- JVMs and Java programs

If you use XPLINK or non-CICS APIs, you must increase the MAXPROCUSER and MAXPROCSYS values. See “Sizing MAXPROCSYS” for guidance.

If CICS is configured to use SSL, you might have to increase the MAXTHREADS and MAXTHREADTASKS values.

If your system uses two or more of these facilities, corresponding further increases in the values for these parameters is appropriate.

### Sizing MAXPROCSYS

The z/OS MAXPROCSYS parameter specifies the maximum number of processes that can be active at the same time, and allows you to manage system resources by limiting the number of processes that the system is to support. If you set the MAXPROCSYS value too low, regions might abend because CICS cannot create a process when attempting to attach a TCB. However, avoid setting the MAXPROCSYS value too high because this value is shared between all the address spaces in a z/OS system.

CICS uses at least two processes for each CICS region, which means that if you have a large number of CICS regions, you must set your system limit to handle these processes. The following table explains which TCBs become processes, depending on the CICS system:

*Table 9. TCBs that become processes*

TCB	Is the TCB always a process?	Description
Jobstep	Yes	The jobstep TCB is always created.
SO	Yes	The SO TCB is always created.

Table 9. TCBs that become processes (continued)

TCB	Is the TCB always a process?	Description
SL	No	The SL TCB is created unless the system initialization parameter, TCPIP=NO, is specified.
QR	No	The QR TCB becomes a process when an OTE TCB is attached (that is; L8, L9, J8, J9, X8, or X9).
SP	No	The SP TCB becomes a process, and the associated S8 TCBs are created and become threads only when SSL is used.
JVM (J8 and J9)	No	The number of J8 or J9 TCBs is controlled by the MAXJVMTCBS SIT parameter. J8 and J9 TCBs running active JVMs require one process each.

In summary, there is always a minimum of two processes for each CICS region, up to a minimum of seven processes for each region, depending on which additional TCBs you are running.

You can issue the following command to give you a list of the processes that are running on your system:

```
D OMVS, A=addressspaceid
```

*addressspaceid* is the address space that you want to query.

If you run this command at system startup, and again when your system has stabilized; for example, when peak J8 and J9 TCBs are running, you can calculate the most appropriate number for MAXPROCSYS.

For more information about changing BPXPRMxx parameters, monitoring system limits and calculating values for system resources, see *z/OS UNIX System Services Planning*.

---

## The console message-handling facility

The console message handling facility is an optional feature of the CICS subsystem that can affect the appearance of CICS messages displayed on an MVS console. It is effective when you specify FORMATMSG=YES as an initialization parameter for the CICS subsystem. The subsystem reformatting is enabled when at least one of the following is executing in the MVS image where the subsystem is defined:

- Any version of CICS Transaction Server
- A message automation subsystem (such as NetView) which enables the MVS subsystem console message broadcasting service

When this facility is used, it affects the messages that are displayed on MVS system consoles in the following ways:

- The subsystem tries to ensure that all console messages issued by all CICS regions have a standard format. The format is:

```
+DFHnnnn  APPLID  MESSAGE-TEXT
```

In this message:

**+DFHnnnn**

– begins in column 1

- The “plus” sign (+) which precedes **DFHnnnn**, is added by MVS to indicate that a problem-state program issued the message. It is not present when CICS issues the message while it is in supervisor state.

#### **APPLID**

- begins in column 13
- The applid inserted into the message is the specific application identifier. This is the identifier that is specified in the system initialization parameter APPLID. It is the only operand when XRF=NO is also specified, or the second operand when XRF=YES is also specified.

#### **MESSAGE-TEXT**

- begins in column 22.

- The subsystem adds routecodes specified in the ROUTECODE subsystem initialization parameter, so the messages might be sent to more destinations.
- The subsystem reformats messages for all CICS releases.
- The subsystem does not reformat messages that are issued by a CICS region that has not yet determined its applid. This includes messages that are issued while processing the system initialization table and its overrides.
- The subsystem routine that reformats the messages does not receive control until after the message has been recorded in the CICS job log. Therefore, the reformatting is not usually apparent in the job log.
- Messages issued by the message domain already contain the applid. The subsystem does not insert the applid into such messages, but it might insert blank characters to cause alignment into standard locations.
- If the original CICS message is a long one, adding the applid might cause the message to exceed the maximum length for an MVS console message. In this case, the original message is suppressed (does not appear on the console), and the reformatted message is issued using the MVS multiple-line console message service to split the message over several lines. Both the original message and perhaps several instances of the reformatted multiple-line message appear in the job log, but only one copy of the reformatted message is displayed on the console.
- For some messages where the applid normally follows a time and date stamp, inserting the applid in the standard position would have resulted in the applid being duplicated within the message. For these messages, the subsystem eliminates the time and date stamp, because these are available from other sources, and only one occurrence of the applied is shown.

## **The DFHSSlly message-formatting initialization member**

You can specify message-formatting initialization parameters for the CICS subsystem in a member DFHSSlly of the SYS1.PARMLIB library, where yy is the suffix that identifies the SYS1.PARMLIB member used to define the CICS subsystem. These parameters, described in this section, are FORMATMSG, HIDEPASSWORD, and ROUTECODES. Code the parameters in columns 1 through 71 of the DFHSSlly member, for example:

```
FORMATMSG=YES,HIDEPASSWORD=YES,ROUTECD= (1,2)
```

or

```
FORMATMSG=YES
HIDEPASSWORD=YES
ROUTECD= (1,2,
3,4,
5,6)
```

**FORMATMSG={YES|NO}**

Specifies whether the CICS applid is to be inserted into all DFH console messages that do not use the CICS message domain.

**YES**

Insert CICS applid into messages.

**NO** Do not insert CICS applid into messages.

**HIDEPASSWORD={YES|NO}**

Specifies whether to mask the password from MODIFY commands used to enter the CICS signon transaction at an MVS console.

**YES**

Mask the password.

**NO** Do not mask the password.

**ROUTECODES=(n1[,n2] . . . .)**

n1, n2... are numbers representing generic routecodes that are added to *all* DFH console messages issued by CICS. The routecodes 1-12 have special meanings:

- 1** Master console action.
- 2** Master console information.
- 3** Tape pool.
- 4** Direct access pool.
- 5** Tape library.
- 6** Disk library.
- 7** Unit record pool.
- 8** Teleprocessing control.
- 9** System security.
- 10** System error/maintenance.
- 11** Programmer information.
- 12** Emulators.

The status of other routecodes is as follows:

**13-20** Available for customer use.

**29-40** Reserved.

**41-128**

Available to authorized programs only.

For more information about these routing codes, see the *z/OS MVS Initialization and Tuning Reference* manual for your version of MVS.

## Default message-formatting initialization parameters

You can define message-formatting initialization parameters for the CICS subsystem in a member DFHSSIyy of the SYS1.PARMLIB library.

To use parameters defined in a DFHSSIyy member other than the DFHSSI00 member, you must specify DFHSSIyy on the IEFSSNaa member in the SYS1.PARMLIB library used to define CICS as an MVS subsystem. If you do not

specify DFHSSlly, the DFHSSIN routine tries to use the parameters that are defined in the DFHSSI00 member. If the DFHSSI00 member does not exist, it uses the default parameters that are defined in the DFHSSIN routine.

If you specify DFHSSlly but it does not exist, the DFHSSIN routine uses the default message-formatting initialization parameters that are defined in the DFHSSIN routine

The default message-formatting initialization parameters defined in the DFHSSIN routine are:

**FORMATMSG=YES,HIDEPASSWORD=YES**  
(generic routecodes are not added to messages)

The default facilities:

- Insert the CICS applid into the CICS console message between the message identifier and the message text. The applid is inserted into only those console messages (starting with DFH) that do not use the CICS message domain. The CICS message domain inserts the CICS applid into all messages that it handles. If the original message is a long one, inserting the CICS applid might cause the message to exceed the maximum length for an MVS console message. In this case, the original message is suppressed (that is, does not appear on the console), and the reformatted message is issued using the MVS multiple-line console message service to split the message text over several lines. Both the original message and perhaps several instances of the reformatted multiple-line message appear in the job log, but only one copy of the reformatted message is displayed on the console.
- Examine each MODIFY command to see if it resembles a MODIFY CICS,CESN ... command. If the MODIFY command contains an old or new password (PS=xxxx,NEWPS=xxxx), the default facilities obliterate the password with asterisks. If the MODIFY command does not contain a password, the password you enter at the MVS console is masked.

**Note:** If your primary subsystem is JES3, the old and new passwords still appear in the JES3 hardcopy log. JES3 records the MODIFY command before the CICS message formatting subsystem can obliterate the password. (This does not happen when the primary subsystem is JES2.) The passwords are suppressed from the console for both JES2 and JES3. For information about the CESN transaction, and about how to prevent passwords from appearing in the hardcopy log, see the *CICS Supplied Transactions* manual.

If you do not specify DFHSSIN in the IEFSSNaa entry that defines CICS, the message handling facilities are not enabled. Also, if you run CICS as a started task, you cannot use the name “CICS” for the procedure name.

## Activating message formatting

After you have defined CICS as an MVS subsystem with support for the console message-handling facility (and have specified the message-formatting parameters in the DFHSSlly member of the SYS1.PARMLIB library), the message-handling facility is activated by the next MVS subsystem to invoke the subsystem console message broadcasting service of MVS console support. This occurs when you start up a supported CICS region (see “The console message-handling facility” on page 103 for a list) or if an automated-operation program, such as NetView, is active in the MVS image.



A newly-started CICS region determines its own applid during initialization. Until the applid is known, the message-formatting facility cannot operate. Therefore, messages issued very early in CICS initialization are not formatted.

## Modules needed to use the console message-handling facilities

To use the console message-handling facilities that are provided by the MVS subsystem functions of CICS, the following CICS modules must be available at MVS IPL time:

### **DFHSEN**

The module that cleans up CICS resources at end-of-memory and at end-of-task.

### **DFHSSGC**

The subsystem generic connect module that connects an active CICS region to the CICS subsystem.

### **DFHSSIN**

The CICS subsystem initialization module.

### **DFHSSMGT**

The subsystem message table that contains the text of messages for the subsystem interface modules.

### **DFHSSWT**

The subsystem interface write-to-operator (WTO) router that determines whether WTO calls should be routed to the appropriate CICS-dependent modules.

These modules must reside in the LPA or in an APF-authorized library in the MVS linklist, as follows:

- The modules DFHSSIN and DFHSSMGT, installed in the *hlq.SDFHLINK* library, must reside in an APF-authorized library in the MVS linklist.
- The DFHSEN module installed in the *hlq.SDFHLPA* library, must reside in the LPA.
- The modules DFHSSGC and DFHSSWT, installed in the *hlq.SDFHLPA* library, must reside either in the LPA or in an APF-authorized library in the MVS linklist.

**Note:** *hlq* is defined by the LINDEXT parameter in the DFHISTAR installation job.

The current versions of these modules are compatible with earlier CICS releases that support console message handling.

For information about adding modules that are installed in the *hlq.SDFHLINK* library to the MVS linklist, see Chapter 10, “Installing CICS-required modules in the MVS linklist,” on page 93.

For information about adding modules installed in the *hlq.SDFHLPA* library to the LPA, see Chapter 15, “Installing CICS modules in the MVS link pack area,” on page 129.

## Coexistence considerations

To use the message-handling facilities for CICS, you should note the following coexistence considerations:

## Automated-operation programs

If your automation system needs to see the console messages before they are reformatted by CICS, its subsystem definition should be placed in IEFSSNXX before the definition for CICS. But if your automation system needs to see the reformatted messages, its definition must come after that of CICS. Consult the documentation of your automation package to determine which applies to you.

## Other CICS releases

If the message-handling facility has been defined to MVS (by the CICS entry in the IEFSSNaa member of the SYS1.PARMLIB library), CICS regions running earlier releases of CICS in the same MVS image have the full benefit of the message handling that has been defined if either of the following is true:

- An automated-operation program, such as NetView, is active in the MVS image.
- A CICS region that supports message handling (see “The console message-handling facility” on page 103 for a list) is running in the same MVS image.

**Note:** A consequence of console messages now having a standard format is that they no longer include date, time and informational messages. If you use these as a token, you must make a change to the code so that it looks for a different token.

---

## EXCI pipe allocation

The external CICS interface is an application programming interface that enables a non-CICS program (a client program) running in MVS to call a program (a server program) running in a CICS region and to pass and receive data by means of a communications area. The CICS application is invoked as though it is linked to by another CICS application program.

This programming interface allows a user to allocate and open sessions or pipes, which operate in "half-duplex", flip-flop" mode, to a CICS region and to pass a distributed program link (DPL) requests over them. The multiregion operation (MRO) facility of CICS interregion communication (IRC) supports these requests, and each pipe maps onto one MRO session, in which the client program represents the sending process and the CICS server region represents the receiving process. There is a default limit of 100 pipes for each EXCI address space.

## The DFHSSlly initialization member

You can specify the EXCI pipe allocation limit for the CICS subsystem in a member DFHSSlly of the SYS1.PARMLIB library, where yy is the suffix that identifies the SYS1.PARMLIB member used to define the CICS subsystem. The parameter, described in this section, is LOGONLIM. Code the parameter in columns 1 through 71 of the DFHSSlly member, for example:

```
LOGONLIM=200
```

### LOGONLIM=nn

The minimum and maximum values that can be specified for nn are 100 and 250.

If the parameter is omitted, or the value specified lies outside the allowed range, then CICS will assume a limit of 100. You should code the parameter only if you want to increase the limit from the default value of 100.

## The EXCI pipe allocation limit

CICS will publish the limit if it is determined during subsystem initialization. It will do so by creating a system level name token pair formatted as follows:

```
Name: input, fixed length 16 byte type
      Bytes 0-7 : The character string 'DFHIRP '
      Bytes 8-15: The character string 'LOGONLIM'
Token: output, fixed length 16 byte type
      Bytes 0-3 : The logon limit, held as fullword binary
      Bytes 4-15: Reserved, set to nulls
```

You can use the callable service, IEANTRT, to retrieve the token. To do this, invoke IEANTRT with level IEANT\_SYSTEM\_LEVEL (EQU 4). The return code is interpreted as follows:

- 0 The name/token pair exists and the token has been retrieved. The logon limit can be extracted from the token.
- 4 The name/token pair does not exist. The logon limit is assumed to be 100.

Any other value indicates that the callable service has detected an error.

## Default EXCI pipe allocation limit initialization parameter

You can define the EXCI pipe allocation limit parameter for the CICS subsystem in a member DFHSSIyy of the SYS1.PARMLIB library.

To use parameters defined in DFHSSIyy member other than the DFHSSI00 member, you must specify DFHSSIyy in the IEFSSNaa member in the SYS1.PARMLIB library used to define CICS as an MVS subsystem.

- If you do not specify DFHSSIyy, the DFHSSIN routine tries to use the parameters that are defined in the DFHSSI00 member.
- If the DFHSSI00 member does not exist, it uses the default parameters that are defined in the DFHSSIN routine.
- If you specify DFHSSIyy but it does not exist, the DFHSSIN routine uses the default parameters that are defined in the DFHSSIN routine.

The default EXCI pipe allocation initialization parameter defined in the DFHSSIN routine is LOGONLIM=100. You should code the parameter only if you want to increase the limit from the default of 100.



---

## Chapter 12. Installing the CICS Type 3 SVC

Install the current level of the CICS Type 3 SVC, DFHCSVC, before you attempt to start a region. To install the CICS Type 3 SVC:

1. Define the DFHCSVC module to MVS. (See “Defining the CICS SVCs to your MVS.”)
2. Install the DFHCSVC module into the LPA.

**Attention:**

**Do not change DFHCSVC attributes.**

Do **not** relink-edit the DFHCSVC module in order to install it into the LPA. The term *install* means move or copy a module into the LPA by using SMP/E, or a copying method that re-blocks the copied modules when the target data set has a smaller block size than the data set you are copying from.

The DFHCSVC module, as supplied, has the attributes AMODE(31) and RMODE(ANY); do **not** change these attributes.

For further information about installing the DFHCSVC module in the LPA, see Chapter 15, “Installing CICS modules in the MVS link pack area,” on page 129.

3. Specify the DFHCSVC number on the CICSSVC system initialization parameter.

The current version of the CICS SVC module is compatible with all earlier releases of CICS, which enables you to run your earlier CICS regions with current regions in the same MVS image.

CICS contains a test to verify that it is using the correct level of the CICS DFHCSVC module. If CICS calls an SVC module using the SVC number specified on the CICSSVC system initialization parameter, and the module is not at the current level, CICS issues message DFHKE0104. As a result of this message, CICS either abends with a system dump, or prompts the operator to enter an alternative SVC number, depending on the option specified on the PARMERR system initialization parameter.

---

## Defining the CICS SVCs to your MVS

You define both the CICS Type 3 SVC and the HPO SVC to your MVS system by specifying SVC Parm statements.

You must define the CICS SVCs in an IEASVCxx member of the SYS1.PARMLIB library, using SVC Parm statements. See the *z/OS MVS Initialization and Tuning Guide* and *z/OS MVS Initialization and Tuning Reference* manuals for a description of the SVC Parm statements. If you are using the default SVC numbers, the CICS entries are as follows:

```
SVC Parm 216,REPLACE,TYPE(3),EPNAME(DFHCSVC)
SVC Parm 215,REPLACE,TYPE(6),EPNAME(DFHHP SVC) [Only required for HPO]
```

For the current SVC modules, you must specify the EPNAME parameters as in the sample CICS entries.

**Note:** If you have a version of the DFHHP SVC module from an earlier release of CICS already link-edited into your MVS nucleus, you do not need to replace it with the latest version. Versions of the DFHHP SVC module from earlier

releases of CICS are compatible with the current release. The CSECT name (EPNAME) of the version of the DFHHP SVC module from earlier releases is IGC215 (or IGCnnn, if SRBSVC=nnn was used as a CICS system generation parameter in the earlier release).

If you are not using the default SVC numbers, change the values 215 and 216 to the SVC numbers you have chosen.

You select the required IEASVCyy member by coding the SVC parameter (SVC=yy) in a SYS1.PARMLIB member (IEASYSxx) which you use to IPL your MVS. When you code new SVC numbers, they do not come into effect until you next IPL your MVS.

---

## Using more than one version of the CICS Type 3 SVC

You may need to use more than one version of the CICS Type 3 SVC, for example to test service applied to the DFHCSVC module while using the current version in a production system.

You can run several CICS regions, at different release levels, in the same MVS image, with each region using its own version of the CICS SVC. However, if some of those regions use MRO, all regions that use MRO must use the latest CICS Type 3 SVC (DFHCSVC module) and the latest DFHIRP module. For information about using the latest SVC with earlier releases of CICS, see “MRO between different CICS releases with a changed SVC number” on page 113 and a pre-Version 3 *Installation Guide*.

To use more than one version of the CICS SVC, rename the new SVC module in the LPA, then respecify the SVC in the SVC Parm statements, as outlined in “Defining the CICS SVCs to your MVS” on page 111. To rename the new CICS SVC module, use the renaming facility of ISPF or IEBCOPY, or the TSO command RENAME, renaming the module to a unique name of your choice. We recommend that you use SMP/E to rename the CICS SVC module in the SDFHLPA library. Use the SMP/E RENAME command to inform SMP/E of the change to the name of the CICS SVC module. Therefore, if you later use SMP/E to apply service to that module, the service is applied to the renamed module in the LPA, **not** the DFHCSVC module.

For example, you might want to use an SVC number 255 for a test CICS region, as well as the default CICS SVC number 216 for your production system:

1. Create and apply an SMP/E USERMOD to RENAME the new CICS SVC module:

```
++USERMOD(umod1) .
++VER(C150) FMID(HCI 640) .
++RENAME (DFHCSVC) TONAME(newname) .
```
2. You could then specify the number 255 for the new CICS SVC version by adding an appropriate statement to the list of SVC Parm statements. That list would then read:

```
SVC Parm 216,REPLACE,TYPE(3),EPNAME(DFHCSVC)
SVC Parm 215,REPLACE,TYPE(6),EPNAME(DFHHP SVC) [Only required for HP0]
SVC Parm 255,REPLACE,TYPE(3),EPNAME(newname) [New CICS SVC version]
```

**Note:** The EPNAME parameter for the new CICS SVC specifies the module name, not the CSECT name, for the new CICS SVC module.

All the SVC Parm statements apply to the same IEASVCxx member of the SYS1.PARMLIB library.

3. Re-IPL MVS to enable all the SVC versions specified in the SVC Parm statements. After you re-IPL MVS, you can use both versions of the CICS SVC, as long as both regions do not use MRO concurrently. If both systems use MRO, only the new, latest, version of the SVC (and the latest DFHIRP module) is used by both regions.
4. In the system initialization table (SIT) for your production system, specify (by the system initialization parameter CICSSVC) the number of the current CICS SVC. Similarly, in the SIT for your test system, specify the number of the new CICS SVC version.

---

## **MRO between different CICS releases with a changed SVC number**

If a CICS TS region, and other CICS regions from earlier releases, in the same MVS image use MRO, all the regions must use the CICS TS SVC module. If when you install the CICS TS SVC in the LPA, you give the SVC a number different from the number defined to the earlier CICS regions, you must respecify the SVC number. On each earlier release CICS region to use the CICS TS SVC, specify the new SVC number on the CICSSVC system initialization parameter.





---

## Chapter 13. Selecting the high-performance option

The high-performance option (HPO) is for users whose top priority is to optimize terminal response times and maximize transaction throughput. HPO improves performance by reducing the transaction path length; that is, the number of instructions needed to service each request.

**Note:** Use of HPO potentially allows CICS application programs to bypass all MVS integrity controls. If you decide to use HPO, ensure that the application programs used on your CICS system meet your own installation's integrity requirements.

The code to support the **VTAM authorized path** feature of HPO (the improved path through VTAM) is in CICS.

---

### Defining DFHHPSVC to MVS

The DFHHPSVC module must be defined to MVS as a Type 6 SVC; the default HPO SVC number defined in the DFHSIT module is 215. If you want to change the default Type 6 SVC number:

1. Define the new number to MVS. (See “Defining the CICS SVCs to your MVS” on page 111.)
2. Define the new number to CICS by using the SRBSVC system initialization parameter.

If you are not using HPO, you should **not** load the DFHHPSVC module into the MVS nucleus. You choose to use HPO explicitly by coding HPO=YES in the system initialization table (SIT).

---

### Loading module DFHHPSVC

Before you can use HPO, ensure that the HPO SVC module is included in the MVS nucleus by one of the following methods:

1. Copy the DFHHPSVC module into SYS1.NUCLEUS, renaming it to IGC215 or the appropriate name if you are not using the default, and specify it on an INCLUDE statement in the NUCLSTxx member of the SYS1.PARMLIB library. (You must also specify the name of the NUCLSTxx member on the NUCLST statement of the LOADxx member of the SYS1.PARMLIB library.) The NUCLSTxx method provides you with greater flexibility in customizing the MVS nucleus than the NMLDEF method described in the method 2.

For further information about coding a NUCLSTxx member, and about a comparison with using the NMLDEF macro, see the *z/OS MVS Initialization and Tuning Guide*.

2. Copy the DFHHPSVC module into SYS1.NUCLEUS and specify it in a nucleus module list (NML) for CICS, created using the NMLDEF macro shown in the sample job in Figure 4 on page 116. This NML selects the CICS members in SYS1.NUCLEUS that are to be loaded into the MVS nucleus, and eliminates the need for the MVS nucleus to be re-link-edited for the DFHHPSVC module (or any other module needed in the MVS nucleus).

For information about coding an NMLDEF macro, see the *z/OS MVS Programming: Authorized Assembler Services Reference Vol 3* manual.

```

//LOADSVC   JOB 'accounting info',MSGCLASS=A,CLASS=A
//NMLDEF    EXEC ASMHCL
//C.SYSIN   DD *
IEANCnnn NMLDEF NUCL=DFHHP SVC
//L.SYSMOD DD DSN=SYS1.NUCLEUS,UNIT=3380,DISP=OLD
//L.SYSIN   DD *
           NAME IEANCnnn
/*
//

```

where *nnn* is the number of the CICS NML, in the range 001 through 256. Choose the value of *nnn* to be unique within your MVS nucleus.

*Figure 4. Sample job stream to load the CICS Type 6 SVC into the MVS nucleus*

## Removing existing DFHHP SVC modules from the MVS nucleus

You can remove a link-edited version of the DFHHP SVC module (for an earlier release of CICS) from the MVS nucleus by running a link-edit job to replace the existing version of the nucleus with one that does not contain the module to be removed.

### Note:

1. If the existing nucleus-resident DFHHP SVC module is known to SMP/E, use the SMP/E UCLIN statement to remove the module entry.
2. You must link-edit the nucleus module, IEANUC0x, with the scatter (SCTR) attribute. If you do not do this, MVS enters a non-restartable wait state at system initialization.
3. If you have a version of the DFHHP SVC module from an earlier release of CICS already installed in your MVS nucleus, you do not need to replace it with the latest version. Versions of the DFHHP SVC module from earlier releases of CICS are compatible with the current release.

---

## Chapter 14. Defining CICS regions as applications to VTAM

To use VTAM terminals with CICS, ensure that your CICS regions are defined to VTAM before you attempt to run them.

To define your CICS regions to VTAM (as VTAM application programs):

1. Define VTAM application program minor nodes for your CICS regions, by specifying APPL definition statements in a member of the SYS1.VTAMLST library (or your own *user.VTAMLST* library). See “Defining specific CICS APPL statements to VTAM.”
2. Issue a VARY ACT command to activate the APPL definitions, and enable the CICS regions to connect to VTAM.
3. Ensure that you have properly defined your VTAM terminals for connection to CICS. This is particularly important if you intend using the CICS autoinstall function. For those terminals for which you want to use autoinstall, code LOGON mode table entries that match the model TYPETERM/TERMINAL definitions that CICS uses. You can either code your own autoinstall models, or use the CICS-supplied model definitions that are generated for you when you initialize the CICS system definition data set (CSD).

For information about defining model and terminal resource definitions to CICS, see the *CICS Resource Definition Guide*.

For programming information about matching VTAM LOGMODE definitions with CICS model definitions, see the *CICS Customization Guide*.

For further information about defining VTAM resources, see the *z/OS Communications Server: SNA Network Implementation* and *z/OS Communications Server: SNA Resource Definition Reference* manuals.

---

### Defining specific CICS APPL statements to VTAM

To define a CICS region to VTAM, specify the minor node name to be used for the CICS region on the VTAM APPL definition statement.

For example, you could use the following definition for the CICS region to be identified as CICSHTH1:

```
*****  
* Specific APPL definition for CICS region CICSHTH1  
*****  
CICSHTH1 APPL AUTH=(ACQ,VPACE,PASS),VPACING=0,EAS=5000,PARSESS=YES X  
          SONSCIP=YES,LUAPFX=XX  
*****
```

**Note:**

1. Code CICSHTH1 on the CICS system initialization parameter APPLID, to define the VTAM application identifier to CICS.
2. See “Naming conventions” on page 219 for information about the naming convention that is used for the CICSHTH1 applid.

### VTAM APPL parameters for CICS regions

When you define your CICS system to ACF/VTAM, include the following parameters on the VTAM APPL statement:

**ACBNAME=acbname**

Specifies the minor node name (*acbname*) that is assigned to this application. This name must be unique within the domain. If you do not specify this parameter, the name of the VTAM APPL statement is taken.

**AUTH=(ACQ,VPACE[,PASS])**

ACQ allows CICS to acquire LUTYPE 6 sessions. VPACE allows pacing of the intersystem flows. You need PASS if you intend using the EXEC CICS ISSUE PASS command to pass existing terminal sessions to other VTAM applications.

**EAS=number**

Specifies the number of network-addressable units. The number must include the total number of parallel sessions for this CICS system.

**HAVAIL=YES**

Indicates that the application supports XRF sessions and can initiate XRF sessions.

**LOGMODE=name**

(For CICS-to-CICS APPC systems.) Defines the name of the MODE table that contains the LU6.2 MODEENT for the secondary sessions.

**LUAPFXstring**

specifies the prefix characters of the LU alias to be assigned when a dynamically generated cross-network CDRSC (with NQNMODE=NQNAME) is created for a session with CICS. VTAM concatenates the characters specified with the next sequential number available to form a VTAM-generated LUALIAS name for the cross-network dynamic CDRSC.

*string*

indicates the two characters to be used as the prefix for all dynamically generated LUALIAS names for dynamic cross-network CDRSCs in session with the CICS region defined by the APPL statement. Remember to take into account the VTAM naming conventions when choosing this prefix. For CICS considerations when specifying the LU alias string, see “Choosing an LUAPFX value” on page 122.

**Note:** VTAM deletes a dynamically-generated LU alias after a terminal session is closed, or the last session of an APPC parallel sessions connection is closed, and the CDRSCTI-specified timeout interval has expired. The permitted range of timeout values is 1 second to 7 days, but generally the default of 8 minutes is acceptable in most situations. The CDRSCTI timer doesn't start until there are no more sessions involving the resource represented by a CDRSC.

For more information about CICS support for the VTAM dynamic LU alias facility, see “VTAM LU alias facility” on page 120.

**PARSESS=YES**

Specifies LUTYPE 6 parallel session support.

**PERSIST=MULTI**

Indicates that the application supports Multi Node Persistent Sessions (MNPS). For further information, see the *z/OS Communications Server: SNA Network Implementation*.

**SONSCIP=YES**

Specifies session outage notification (SON) support. SON enables CICS, in certain cases, to recover a session after session failure without operator intervention.

**VPACING=number**

Specifies the maximum number of normal-flow requests that another logical unit can send on an intersystem session before waiting to receive a pacing response. Start with a value of 5.

## VTAM version and release level indicator

The terminal control modules in CICS are assembled against Communication Server. You can use any release of Communication Server VTAM. For details of the minimum level of products that you can use with the current release, see Software Requirements in the *CICS Transaction Server for z/OS Release Guide*.

CICS can communicate with different levels of VTAM. It can find out which level you are using, and hence what level of function is available. This means that you can upgrade CICS and SecureWay™ Communication Server VTAM at different times. CICS finds out whether extra function is available when a new version of VTAM is installed, and produces a message if the facilities are not being exploited fully.

## Message DFHZC3473 on opening the VTAM ACB

If the master terminal operator opens the VTAM ACB for the first time, using the command CEMT SET VTAM OPEN, but CICS is not using all available VTAM function, message DFHZC3473 is sent to the transient data destination called CSNE. The same message is sent there if the ACB is opened automatically during initialization, rather than by CEMT.

## Cross-domain considerations

If you want to use VTAM services to access a CICS region on another MVS image, you must ensure that the required cross-domain services are defined to the VTAMs involved.

For example, to be able to use a VTAM APPC connection between a CICS region (applid CICSHTH1) on MVS image MVSH and a CICS region (applid CICSHAJ1) on MVS image MVSJ:

1. Define the cross-domain services (CDRSC) for accessing CICSHAJ1 in a member of the SYS1.VTAMLST library (or your own user.VTAMLST library) for MVSH.
2. Issue a VARY ACT command on MVSH to activate the CDRSC definition for accessing CICSHAJ1.
3. Define the cross-domain services (CDRSC) for accessing CICSHTH1 in a member of the SYS1.VTAMLST library (or your own user.VTAMLST library) for MVSJ.
4. Issue a VARY ACT command on MVSJ to activate the CDRSC definition for accessing CICSHTH1.

For example, you could:

1. Create the following CDRSC definition in a member of the VTAMLST library on MVSH:

```

CDIDHAJ1 VBUILD TYPE=CDRSC
*****
* CDRSC for access to applid CICSHAJ1 on MVSJ
*****
CICSHAJ1 CDRSC CDRM=IYAMCDRM MVSJ

```

2. Issue the following command on MVSH to activate the cross-domain services to CICS SHAJ1 on MVSJ:  

```
/V NET,ACT,ID=CDIDHAJ1
```
3. Create the following CDRSC definition in a member of the VTAMLST library on MVSJ:  

```
CDIDH1 VBUILD TYPE=CDRSC
*****
* CDRSC for access to applid CICSHTH1 on MVSH
*****
CICSHTH1 CDRSC CDRM=IYALCDRM MVSH
```
4. Issue the following command on MVSJ to activate the cross-domain services to CICSHTH1 on MVSH:  

```
/V NET,ACT,ID=CDIDH1
```

## VTAM LU alias facility

Specifying a prefix string on the LUAPFX parameter of the CICS APPL statement indicates that VTAM is to generate LU ALIAS names for dynamic cross-network CDRSCs in session with the CICS region defined by the APPL statement. This enables CICS to use an LU alias for autoinstalled terminals and work stations and thus ensure unique names in a CICSplex comprising terminal-owning and application-owning regions (TORs and AORs). VTAM generates the LU ALIAS names dynamically.

CICS supports both forms of the VTAM alias function—predefined and dynamic—only where shown in the following table:

	CICS-to-CICS APPC connections (APPL definitions)		APPC devices (LU definitions)				Terminals	
	Synclevel 1	Synclevel 2	Synclevel 1		Synclevel 2			
	Predefined alias only		Predefined alias	Dynamic alias	Predefined alias	Dynamic alias	Predefined alias	Dynamic alias
VTAM	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
CICS	Yes	No	Yes	Yes	No	No	Yes	Yes

### Note:

1. The LU alias is used as the NETNAME for terminals and work stations that logon to a CICS region.
2. CICS does not support LU alias for synclevel 2 connections (LUTYPE 6.1 and 6.2) and ignores any LU alias for these LU types, and continues to use the network name defined in the VTAM APPL statement.

### Dynamic LU alias support

CICS supports the use of a dynamic LU alias for CICS terminals and workstations that are autoinstalled only. You enable dynamic LU alias support by specifying LUAPFX on the VTAM APPL definition for any CICS terminal-owning region that could receive duplicate netnames. Also, when starting VTAM, specify the following options on the VTAM START command:

- NQNMODE=NQNAME
- CDRSCTI=*n* to specify the length of time that the session name should last after the last session has logged off.

**Note:**

1. Make the time specified on CDRSCTI long enough to cover any time interval specified on CICS START commands that are issued against a terminal resource that uses a dynamic LU alias. This applies to STARTS with a delay that run on both a TOR or AOR. If the CDRSCTI time is not long enough, a resource could log off and then log back on again with a different network name and thus a different TERMID.
2. The CDRSCTI time interval should also be greater than that specified on the CICS AILDELAY system initialization parameter. However, if your applications have no dependency on the network name or termid, you can disregard CDRSCTI or set it to 1.

VTAM generates a dynamic LU alias only if LUAPFX is specified on the CICS APPL statement and the resource comes from another network. That is, it has a different network name from the network to which the CICS region belongs.

**When to use dynamic LU alias:** Use dynamic LU alias where:

- Your cross-network terminals and workstations that logon to CICS are mainly autoinstalled.

The CICS region receives logons from terminals and synclevel 1 connections (both parallel and single sessions) and those logons (or binds) are from cross-network resources that might have duplicate network names.

However, be aware that synclevel 1 connections could become synclevel 2 in the future. For example, if you have a connection between a TXSeries and CICS TS it is synclevel 1, but if you change to using TXSeries with a PPC gateway, synclevel 2 is used. CICS does not support dynamic LU aliases for synclevel 2 APPC connections.

- An AOR receives shipped terminals or connections with duplicate network names from different TORs.

**Predefined LU alias support**

CICS supports the use of a predefined LU alias for CICS terminals and workstations that are explicitly defined and those that are autoinstalled. You can also use a predefined LU alias for CICS regions that communicate using CICS intersystem communication (ISC). You enable predefined alias support by specifying LUALIAS=*alias* on any cross-domain resource (CDRSC) that needs a specific alias.

**Note:** A terminal or APPC sync level 1 work station that is defined to CICS on an explicit resource definition (that is, it is not autoinstalled) and is in a different network, requires a CDRSC definition with a specific alias on the LUALIAS parameter. This overrides the dynamic generation of an alias where LUAPFX is specified on the CICS region's APPL statement. To ensure that CICS can match the VTAM LU alias with the installed terminal definition, the LUALIAS value must match the NETNAME specified on the CICS TERMINAL resource definition.

An LUALIAS option in the CDRSC is effective if the resource comes from another VTAM domain (or network). That is, it is not used if the resource comes from the same MVS image, but is used if the resource comes from another MVS image regardless of whether it is from the same sysplex, another sysplex in the same network, or from a different sysplex. If an LU alias is predefined, a dynamic LU alias is not generated.

**When to use predefined LU alias:** Use predefined LU alias where:

- Dynamic LU alias is in operation in a CICS region and your terminals or workstations are explicitly defined on CICS terminal resource definitions with explicit terminal identifiers. In this case, you use predefined LU aliases to override the generation of dynamic LU aliases, which CICS would fail to match with any installed resource definition.
- Dynamic LU alias is not in operation in a CICS region, to avoid any conflict with duplicate network names.

**Cross-network devices that need predefined LU alias:** If the following VTAM cross-network resources are to be connected to a CICS region that is defined to VTAM with LUAPFX specified on its APPL statement, they must each have a CDRSC LUALIAS=netname entry:

- CICS RDO-defined terminals connected from another network. These include VTAM terminals that cannot be autoinstalled:
  - Pipeline terminals
  - Automatic teller machines (3614 and 3624)
  - Devices for which CICS does not receive logons, such as printers.
- LUTYPE 6.2 synclevel 1 connections that may be bound using limited resources. Like other LUTYPE 6.2 connections limited resource connections release their dynamic LU alias when CDRSCTI expires after the last session is unbound. However, these sessions are unbound whenever they are not in use, and if they rebind after the dynamic LU alias is released, CICS would install another connection, potentially with a different LU alias.
- CICS RDO-defined work stations (LUTYPE 6.2 synclevel 1 connections) connected from another network.
- Resources that require an LU name in a RACF profile definition, or resources for which prior knowledge of the LU name is required.

### Choosing an LUAPFX value

When choosing an LUAPFX value, consider the scope of this parameter within the CICSplex, and also consider its scope within the sysplex in which your CICS regions operate.

A predefined LUALIAS name is supplied to CICS for cross-domain and cross-network resources. All the CICS regions in an MVS image share the same VTAM and are in the same domain. A CICS region in a different MVS image uses a different VTAM and is thus in a different domain. Resources coming from one VTAM to another, but which share the name NETID, are cross-domain resources

A dynamic LUALIAS name is only supplied to CICS for cross-network resources. A resource is a cross-network resource if it has a different network id (NETID). VTAM ensures that all the dynamic LUALIAS names assigned in one MVS image are unique. However, CICS needs network names to be unique across MVS images so that we do not get network name clashes in AORs.

It is important that all CICS regions across all connected networks use unique APPLIDs. This is true whether or not dynamic LUALIASs are used—it just makes it more important.

To ensure that all VTAM resources in a CICSplex have unique network names, use the LUAPFX prefix as follows:

- Specify LUAPFX on terminal-owning regions (TORs) only.



- Use the same LUAPFX value for all the CICS TORs in the same MVS image (that is, for all the TORs that are connected to the same VTAM), but ensure the LUAPFX is different in each MVS image in the sysplex..

If the LUAPFX values are not the same throughout an MVS image, you risk one resource having two different network names in the CICS regions in that image.

If the LUAPFX values are not unique to each MVS image in the sysplex, you risk two resources attempting to install in a TOR with the same dynamic LUALIAS, or having two resources with the same network name in an AOR.

To ensure the uniqueness of the LU prefix in each MVS, IBM recommends that you use model APPL definitions, and within these use an MVS system symbol (&SYSC clone) as suggested in the *z/OS Communications Server: SNA Resource Definition Reference*.

**Note:** If you use VTAM generic resources and your CICS TORs are spread across different MVS images, be aware that if a resource with a dynamically allocated LU alias logs off and then logs on again, and VTAM switches the resource to a VTAM in another MVS image, a different LUALIAS is assigned because of the different LUAPFX value.

- Avoid using an LUAPFX value that corresponds to the first two characters of CICS RDO-defined terminal names or connection names installed in the CICSplex.

### Other considerations when using LU aliases

The following are some other factors to consider when you are planning to use VTAM LU aliases with CDRSC resources:

#### Predictable termids

If you need autoinstalled terminal resources to have a predictable and reproducible TERMID for such things as temporary storage queue names and START requests, you may need to modify your autoinstall user-replaceable module (URM) to select a reproducible TERMID from the network qualified name (NQNAME) supplied in the CINIT or the BIND.

There is an example of such code (commented-out) in the sample autoinstall URM, which extracts the network qualified name from the CINIT and BIND. The example illustrates how to create a TERMID from the last non-blank character of the NETID and the last 3 non-blank characters of the *real* network name (NETNAME).

#### MVS workload management

If your MVS workload policies specify LU name classifications, remove the LU name for any cross-network resources that are autoinstalled by CICS.

#### Recovery and persistent sessions support

Resources for which CICS uses any VTAM LU alias (predefined or dynamic) and which come from a different network are not cataloged by a CICS region that is not using persistent session. This means the terminal sessions for the resources cannot be recovered during an emergency restart.

Resources for which CICS uses any VTAM LUALIAS (predefined or dynamic) and which come from a different network are cataloged if CICS is using persistent sessions. This enables CICS to restore resource terminal session information from the CICS catalog pending recovery of the session from VTAM. However, if the resource does not persist, the resource is deleted during an emergency restart.

This action is necessary because VTAM may have been restarted, which would cause dynamic LU aliases to be reissued to different sessions. CICS is unable to tell if VTAM has been restarted, and CICS cannot tell the difference between a predefined and a dynamic LU alias.

#### **CLSDST PASS**

If you ISSUE PASS (CLSDST PASS) for a terminal that uses a dynamic LU alias to pass control to another CICS region in another MVS image, the resource is known by a different network name in the receiving CICS. This is true if the APPL statement of only one or both the CICS regions specify LUAPFX to activate dynamic LU alias.

#### **Generic resources**

If a number of generic resource TORs are in two different MVS images, a terminal or work station that logs on to one image is assigned a different network name if it logs off and logs on to a TOR in another image.

#### **FEPI**

FEPI front end systems are not supported by VTAM LU alias.

---

## **Defining VTAM requirements for CICSplex SM**

ACF/VTAM definitions are required to identify each CMAS used by CICSplex SM. This involves creating VTAM application definitions and, optionally, cross-domain resource management definitions.

To create VTAM application definitions and cross-domain resource management definitions for a CMAS, you must perform the following steps:

1. Create a VTAM application definition for each CMAS you will be using.
2. Define each CMAS as a cross-domain resource.
3. Add the application and cross-domain resource definitions to the VTAM configuration list.
4. Activate the definitions.

#### **Note:**

1. Before you perform these steps, be sure to specify the size of the VTAM buffers.
  - For the VTAM-to-NCP connection, specify
    - MAXDATA  $\geq$  4096
  - For the NCP-to-VTAM connection, specify
    - MAXBFRU \* IOBUF  $\geq$  4096
    - MAXBFRU \* UNITSZ  $\geq$  4096
  - For the NCP-to-NCP connection, specify
    - TRANSFR \* BFRS = RUSIZE  $\geq$  4096

The size specified should be 36 bytes less than the smallest MAXDATA value in any NCP through which the link may pass. The 36 bytes allow for the headers that are required for VTAM. For more information about the requirements for the VTAM-to-NCP connection, refer to the *VTAM Resource Definition Reference* manual for your level of VTAM. For more information about the requirements for the NCP-to-VTAM and the NCP-to-NCP connections, refer to the *NCP Resource Definition Reference* manual for your level of NCP.

If you need help determining or modifying your VTAM buffer specifications, confer with the VTAM system programmer at your enterprise.

2. Depending on your VTAM conventions, you may need to modify the procedures that are described in this section. Specifically:
  - Change references to the SYS1.VTAMLST library if you do not keep your definitions in the default VTAM list.
  - Modify the APPL and CDRSC statements if you want to add these statements to existing members, rather than create new ones.

## Step 1: Creating a VTAM application definition for a CMAS

To establish a VTAM application definition for a CMAS, either create a new member (*major node*) or access an existing member in the SYS1.VTAMLST library. Then add the following APPL statement:

```

          VBUILD TYPE=APPL
name     APPL ACBNAME=acbname,AUTH=(VPACE,ACQ,SPO,PASS),           x
          EAS=10,PARSESS=YES,SONSCIP=YES,APPC=NO,                 x
          VPACING=number
  
```

where:

**name** Is a 1- to 8-character unique name.

**acbname**

Is the node name of this CMAS. This name must be unique within the domain. If you omit this parameter, the name of the VTAM APPL statement is used.

**vpacing**

Is the maximum number of normal-flow requests that another logical unit can send on an intersystem session before waiting to receive a pacing response. Start with a value of 5.

For example, to create a VTAM application definition for the CMAS on SYSA, you might create a member (APCMAS1) in the SYS1.VTAMLST library that contains the APPL statement:

```

          VBUILD TYPE=APPL
CMS1     APPL ACBNAME=CMS1,AUTH=(VPACE,ACQ,SPO,PASS),           x
          EAS=10,PARSESS=YES,SONSCIP=YES,APPC=NO,                 x
          VPACING=5
  
```

The same type of definition is needed for each CMAS you use.

## Step 2: Defining cross-domain resources for a CMAS

You should define cross-domain resources (CDRSCs) when:

- A CMAS that is to communicate with another CMAS cannot take advantage of adjacent CDRSCs.
- You want to minimize the overhead involved in using adjacent CDRSCs.

To establish a CDRSC definition, you must either create a new member or access an existing member in the SYS1.VTAMLST library. In the new or existing member, specify the following CDRSC statement for each CMAS that you want to communicate with:

```

          VBUILD TYPE=CDRSC
name     CDRSC CDRM=cdrm
  
```

where:

- name** Is the name you assigned to a CMAS in Step 1.
- cdrm** Is the name of the MVS image previously identified as the cross-domain resource manager (CDRM).

For example, to allow the CMAS on SYSA to communicate with the CMASs on SYSB and SYSC, you might create the member CDRCMS1, in the SYS1.VTAMLST library, which contains the CDRSC statements:

```
          VBUILD TYPE=CDRSC
CMS2     CDRSC CDRM=VTAMB
CMS3     CDRSC CDRM=VTAMC
```

where VTAMB and VTAMC are the cross-domain resource manager names that are assigned to SYSB and SYSC respectively.

The same types of definitions are also needed for the CMASs on SYSB and SYSC. That is, for the CMAS on SYSB, you could create a member (CDRCMS2), which contains the CDRSC statements:

```
          VBUILD TYPE=CDRSC
CMS1     CDRSC CDRM=VTAMA
CMS3     CDRSC CDRM=VTAMC
```

For additional information about cross-domain resources, see the *VTAM Resource Definition Reference* manual.

### Step 3: Updating the configuration list for a CMAS

If, in Step 1 or 2, you created new members in the SYS1.VTAMLST library, you must update the VTAM configuration list for each MVS image. When VTAM starts, it automatically activates the new members.

To do this, add the new member names to the end of the configuration list in the appropriate ATCCONxx member of the SYS1.VTAMLST library. To find the suffix of the ATCCONxx member:

- Get the suffix of the COMMNDxx member from the CMD= parameter in the IEASYSxx member in SYS1.PARMLIB.
- Get the suffix of the ATCSTRxx member from the LIST= parameter on the command that is used to start VTAM in the COMMNDxx member in SYS1.PARMLIB. If you do not start VTAM from the COMMNDxx member, get the suffix from the LIST= parameter of the command that you use to start VTAM.
- Get the suffix of the ATCCONxx member from the CONFIG= parameter in the ATCSTRxx member in SYS1.VTAMLST.

To illustrate, the examples shown in Steps 1 and 2 assume that the members APCMAS1 and CDRCMS1 exist. To add these members to the end of the configuration list in ATCCONxx, you would specify:

```
APCMAS1,                                     x
CDRCMS1
```

**Note:** If you added the CMAS and cross-domain definitions to existing members, ATCCONxx should already contain these member names.

### Step 4: Activating the major nodes for a CMAS

You can activate the definitions that are created in Steps 1 and 2 by either restarting VTAM for each system, or manually activating the definitions.

To manually activate a major node, you can issue the following commands, where name identifies a major mode that was created (or modified) in Steps 1 and 2:

- Deactivate the major node if it is currently active by issuing the command:

```
VARY NET,INACT,ID=name
```

- Activate (or reactivate) the major node by issuing the command:

```
VARY NET,ACT,ID=name
```

To ensure that the major node has been activated, issue the command:

```
D NET,ID=name
```

For example, to activate the member APCMAS1 and then ensure that it has been activated, you would issue the commands:

```
VARY NET,INACT,ID=APCMAS1
```

```
VARY NET,ACT,ID=APCMAS1
```

```
D NET,ID=APCMAS1
```

The preceding steps need to be performed for each CMAS you may be using.



---

## Chapter 15. Installing CICS modules in the MVS link pack area

This section describes:

- What you should consider before you install modules in the MVS link pack area.
- What you should do to use CICS modules in the MVS link pack area:
  - MVS link pack area
  - “Defining the CICS LPA library to your MVS” on page 133
  - “Installing CICS modules in the LPA” on page 134
  - MVS link pack area

---

### Preparing to install CICS modules in the MVS link pack area

Before you install modules in the MVS link pack area, you should consider the following points, described in subsequent topics:

- “Benefits of using the MVS link pack area”
- “What is meant by the MVS link pack area?”
- “Which modules must be in the MVS link pack area?” on page 130
- “Which modules can be in the MVS link pack area?” on page 131
- “Service considerations” on page 132

### Benefits of using the MVS link pack area

The benefits of placing code in the MVS link pack area are:

- It is protected from possible corruption by user applications. Because the MVS link pack area is in protected storage, it is virtually impossible to modify the contents of these programs.
- Performance can be improved, and the demand for real storage reduced, if you use the MVS link pack area for program modules. If more than one copy of the same release of CICS is running in multiple address spaces of the same processor, each address space requires access to the CICS nucleus modules. These modules may either be loaded into each of the address spaces or shared in the MVS link pack area. If they are shared in the MVS link pack area, this can reduce the working set and therefore, the demand for real storage (paging).
- You can decrease the storage requirement in the private area by judicious allocation of the unused storage in the MVS link pack area created by rounding to the next segment.

If you know the amount of space that you need in the LPA, and from that the total size of the MVS common area above the CICS private storage, you can determine which 1MB segment the boundary between the two areas lies on. This may indicate that there is some space in the MVS common area that is left unused, which you could use for CICS LPA-eligible modules. By moving more modules from CICS private storage to the LPA, you decrease the space that is needed for modules in CICS private storage.

### What is meant by the MVS link pack area?

The MVS link pack area comprises several areas, both above and below 16 MB. In these topics, the term **MVS link pack area** refers to the pageable link pack areas above and below 16 MB where modules that are used from the MVS link pack area are usually installed.

**Note:** The MVS link pack area has both pageable and fixed parts. Although you can install CICS modules into the fixed parts, we recommend that you use the pageable areas for performance reasons.

The term **LPA** specifically refers to the MVS link pack area below 16 MB, and the term **ELPA** specifically refers to the area above 16 MB. A module that is link-edited with the RMODE(ANY) attribute is loaded into the ELPA.

If you install a module into the LPA or ELPA, that module will not be retrieved from the MVS link pack area until you re-IPL your MVS with CLPA specified. To use the new module and avoid an IPL, you can use the MVS dynamic LPA in one of the following ways:

- With the SETPROG LPA command
- With an LPA statement in a PROGxx member of PARMLIB, which specifies the SET PROG=xx command

Note that dynamic LPA must not be used with DFHIRP.

## Which modules must be in the MVS link pack area?

The CICS modules listed in Table 10 must be in the MVS link pack area for the reasons that are given in the notes after the table.

*Table 10. CICS modules required in the MVS link pack area*

Module	Description	When needed in LPA	See notes after this table
DFHCSVC	CICS Type 3 SVC	Always	1, 2, 3,4, and 6
DFHDSPEX	CICS post exit stub	Always	1 , 3, and 5
DFHDUMPX	SDUMPX IEASDUMP QUERY exit	Always	1, and 3
DFHIRP	Interregion communication program	To use MRO, CICS shared database, or the console message-handling facility	1, 2, 3, and 6
DFHSSSEN	Subsystem interface end-of-memory / end-of-task clean up routine	To use the console message-handling facility	1, 2, 3, and 6
DFHSSGC	Subsystem generic connect module	To use the console message-handling facility	3, and 7
DFHSSWT	Subsystem interface WTO router	To use the console message-handling facility	3, and 7
DFH99SVC	Dynamic allocation - SVC services	Always	1, and 3

**Note:**

1. Can be used only from the MVS link pack area, and must be installed there before CICS can be started.
2. You must always install the latest service level of the modules DFHCSVC, DFHIRP (if needed) and DFHSSSEN.
3. The version of this module that is supplied with the current release, is downward-compatible with earlier releases of CICS. It works correctly



with CICS regions running earlier releases. Therefore, if you are running different releases of CICS on the same MVS image, you should use the latest version of this module.

4. The DFHCSVC module must be defined to MVS as a Type 3 SVC (default SVC number is 216), and if you use a non-default SVC number, you must define it to CICS on the CICSSVC system initialization parameter.

#### **Moving DFHCSVC into the:**

Do **not** use the linkage editor to install the CICS SVC module into a library in the MVS link pack area. To copy or move the module from the *hlq.SDFHAUTH* library to the nominated library in the MVS link pack area, you should use either a suitable copy utility program, such as IEBCOPY, or an SMP/E USERMOD with ++MOVE statements.

5. If you are running earlier releases of CICS with the latest version, you must ensure that the latest version of the DFHDSPEX module is installed in the MVS link pack area. The DFHDSPEX module must be in the MVS link pack area for integrity reasons, but the post exit routine, DFHDSAUT, can reside either in the MVS link pack area, or in the CICS address space. This enables you to use different levels of the DFHDSAUT module in different CICS regions running in the same MVS image, because the DFHDSAUT module may not be compatible between CICS versions.
6. To communicate by MRO, all CICS regions in the same MVS image must use the latest level of the modules DFHCSVC, DFHIRP, and DFHSEN in the MVS link pack area.  
  
If a region detects that DFHIRP is at a lower level, when attempting to open interregion communication, it issues message DFHIR3799, and interregion communication fails to open.
7. To use the console message-handling facility, these modules must reside either in the MVS link pack area or in an APF-authorized library in the MVS linklist.

## **Which modules can be in the MVS link pack area?**

Besides those CICS modules that must be in the MVS link pack area, other CICS modules and user application program modules are available from the MVS link pack area.

### **CICS modules**

A CICS module optionally installed in the MVS link pack area (that is, not a module required in the MVS link pack area) can be used only by the release of CICS to which it relates.

Those CICS modules that can reside above 16MB (for example, the CICS message table, DFHMGT), are loaded above 16MB. Such modules can also be installed in the extended link pack area (ELPA).

CICS modules eligible to be used from the MVS link pack area are listed in the CICS-supplied USERMODs, DFH\$UMOD (for base CICS modules), which is in the *hlq.SDFHSAMP* library. Details of LPA-eligible modules are in Table 28 on page 375, to help you select those CICS modules that you want to install in the MVS link pack area.

## User application programs

User application programs can be used from the MVS link pack area if they are read-only and:

- Written in COBOL, do not overwrite WORKING STORAGE, and are compiled using VS COBOL II, or later version. (The CICS translator generates a CBL statement with the required compiler options.)
- Written in PL/I (do not overwrite STATIC storage) and compiled using one of the versions of PL/I listed in High-level language support in the *CICS Transaction Server for z/OS Release Guide* or later. (The CICS translator inserts the required REENTRANT option into the PROCEDURE statement.)
- Written in C/370, compiled with the RENT option, and link-edited with the RENT option.
- Written in Assembler language, assembled with the RENT option, and link-edited with the RENT and REFR options.

Command-level user application programs compiled using a Language Environment-conforming compiler, or written in Assembler language or C/370, may be loaded above 16MB. (For information about installing application programs, see [../com.ibm.cics.ts.applicationprogramming.doc/topics/dfhp3\\_installprog\\_intro.dita](#) in the *CICS Application Programming Guide*.)

A read-only module that may reside above 16MB is also eligible for the extended link pack area (ELPA).

## Service considerations

Using modules with mismatching service levels can cause unpredictable results. To be safe, do not use the LPA version of a module if it differs from the version in the CICS libraries that you are using.

Load modules used from the LPA might be at a lower service level than the rest of your CICS region in any of the following circumstances:

- You are running CICS from libraries which belong to a target zone currently at a higher service level than the LPA zone.
- You have applied service to the LPA zone since the last IPL of MVS.
- You are not using the MLPA to replace service-updated load modules, but have applied service to the LPA zone since last IPL of MVS for which CLPA (create link pack area) was specified.

Thus, if you have applied service to a load module in your CICS libraries, you should also apply the service to the LPA version of the module, if one exists. This stipulation is there so that the MVS link pack area always contains tested load modules.

Use the SMP/E RESTORE function to back off the USERMOD before the LPA zone is updated or copied. Then apply the USERMOD again.

If you have used a copy of the CICS-supplied USERMODs to install modules into the MVS link pack area, and the original USERMOD is serviced, you may like to reflect the changes in your version of the USERMOD.

---

## How to use modules in the MVS link pack area

To use CICS modules in the MVS link pack area:

1. Check that you have enough space for the selected modules.
2. Install the modules in the MVS link pack area.
3. Control the usage of modules from the MVS link pack area.

---

## Space requirements for CICS modules in the MVS link pack area

Allow enough space in the MVS link pack area for you to install those CICS modules that you intend using from there. You can find out how much space you need by:

- Reviewing the sizes of the modules that you want to install in the MVS link pack area, as given in Table 28 on page 375.
- Reviewing the module index of a system dump for the CICS region started with the system initialization parameter LPA=NO.
- Calculating the module sizes that are given for each module in the listing of modules that are provided by the IEHLIST utility program.

Remember also to allow space for any of your user application programs that you intend using from the MVS link pack area.

**Note:** The total space needed depends on how the operating system packages the modules into the MVS link pack area.

### What next?

Once you have determined the space needed in the MVS link pack area, you must next create a library with enough space and define it to your MVS. This is in the topic “Defining the CICS LPA library to your MVS.”

---

## Defining the CICS LPA library to your MVS

CICS supplies the library *hlq.SDFHLPA*. This library contains the modules that must be in the LPA. You can also use this library to install other CICS modules or application programs which you want to use from the LPA.

You can give the *hlq.SDFHLPA* library your own index, but if you do, you must specify the new index on the LINDEX parameter of the DFHISTAR job.

Add the full name of the *hlq.SDFHLPA* library to an LPALSTxx member of SYS1.PARMLIB. This ensures that the library contents are loaded into the PLPA at the next IPL of your system when CLPA is specified. Also APF-authorize the *hlq.SDFHLPA* library, by adding its name to an IEAAPFxx member of the SYS1.PARMLIB library.

For more information about this see “Verifying your Java components installation” on page 236.

You should also RACF-protect the *hlq.SDFHLPA* library, to prevent unauthorized or accidental modification of this library. For information about protecting the CICS libraries, see the *CICS RACF Security Guide*.

### What next?

During migration to the current release use of the MVS link pack area, you may like to add a DD statement for the *hlq.SDFHLPA* library to the DFHRPL concatenation of your CICS startup job stream.

You can install into the *hlq.SDFHLPA* library the CICS modules to be used from the MVS link pack area. This is in “Installing CICS modules in the LPA.”

---

## Installing CICS modules in the LPA

By *install*, we mean move or copy a module into a suitable LPA library, by using SMP/E, or by using a copying method that re-blocks the copied modules when the target data set has a smaller block size than the data set you are copying from (for example, use the COPYMOD function of the IEBCOPY program). A procedure for installing modules into the MVS link pack area by using SMP/E is in this section.

You should not relink-edit the modules in order to get them into the LPA library. CICS modules as supplied have the necessary attributes that cause MVS to load them automatically above the line (into the ELPA).

To install modules in the CICS LPA library, and to ensure that SMP/E can continue to service them, complete the following steps:

1. Select those modules that you want to use from the MVS link pack area, and specify them in the SMP/E USERMOD to be used to install the modules in the MVS link pack area.

You can use the CICS-supplied USERMOD, LPAMODS, or create and use your own version.

If you use your own version of a USERMOD, this can include ++MOVE statements from both CICS-supplied USERMODs.

2. Receive the USERMOD into the CICS global zone.
3. Apply the USERMOD to the LPA zone.

**Note:** When you have installed all your modules into the CICS LPA library (and defined it to MVS), you should re-IPL your MVS with CLPA specified to enable the modules to be used from the CICS LPA library.

These steps are in the following sections.

## Selecting modules for the MVS link pack area

You should install in the MVS link pack area only those modules that you want to use from the MVS link pack area. Table 28 on page 375 lists the CICS-supplied modules eligible for the MVS link pack area, and gives descriptions and other information to help you select those CICS modules that you want to use from the MVS link pack area.

To install modules in the MVS link pack area, you should use an SMP/E USERMOD that contains ++MOVE statements for only the modules to be installed in the MVS link pack area.

## The CICS-supplied SMP/E USERMOD, DFHUMOD

CICS supplies an SMP/E USERMOD called DFHUMOD (in member DFH\$UMOD in the *hlq.SDFHSAMP* library). This USERMOD contains ++MOVE statements for all CICS modules, in the *hlq.SDFHAUTH* and *hlq.SDFHLOAD* libraries, that are eligible for the MVS link pack area. The USERMOD also indicates whether each module is LPA- or ELPA-eligible. You can choose which of the modules to install in the MVS link pack area by creating your own version of the USERMOD. Your selection is generally include modules in the working set of the installation.

## Changing a CICS-supplied USERMOD

If you intend changing a CICS-supplied USERMOD, to choose modules to install in the MVS link pack area, take a copy of the USERMOD and update the copy only. If you have copied the *hlq.SDFHSAMP* library, for instance when changing user-replaceable programs, then you already have copies of the CICS-supplied USERMODs. If the original *hlq.SDFHSAMP* library is serviced, and the USERMOD is modified, you may like to reflect the changes in your version.

## Preparing the USERMOD

To choose which read-only modules to install in the MVS link pack area, edit your copy of the SMP/E USERMOD to:

1. Comment out the ++MOVE statements for the modules that you do not want to install in the LPA, and
2. Move the remaining ++MOVE statements (for the modules that you do want to install in the LPA) one column to the left, so that the ++MOVE statements start in column one of the USERMOD module.
3. Add ++MOVE statements for your user application program modules that you want to install in the LPA, with the ++MOVE statements starting in column one of the DFH\$UMOD module.

## Receiving and applying the USERMOD

Receive the USERMOD into the CICS global zone and apply it to the LPA target zone. This causes SMP/E to move those load modules you have specified from the named CICS target library (*hlq.SDFHLOAD* or *hlq.SDFHAUTH*) into the CICS LPA library. Applying the USERMOD also updates the corresponding LMOD entries within the target zone SMPCSI.

Do not accept the USERMOD into the distribution zone, and for the time being, do not apply it to any other target zone.

To receive and apply the CICS-supplied sample usermods in DFH\$UMOD, you can use the associated job DFHLPUMD which is tailored to your CICS environment and stored in the *hlq.XDFHINST* library when you run the DFHISTAR job.

### What next?

To enable CICS to use the modules that you have installed in the MVS link pack area, re-IPL your MVS with CLPA specified.

Also specify to CICS that it is to use modules from the MVS link pack area. You can also control which modules are used from the MVS link pack area in several ways. This is in the topic MVS link pack area.

---

## Controlling the use of modules from the MVS link pack area

This topic describes what you must do to enable CICS to use modules from the MVS link pack area, and what you can do to specify that CICS **is not** to use eligible modules from the MVS link pack area.

The methods for controlling the use of modules from the MVS link pack area **do not** apply to the modules DFHCSVC, DFHDSPEX, and DFHIRP. These modules can be used only from the MVS link pack area.

### Modules in the MVS link pack area from hlq.SDFHAUTH

CICS uses standard MVS load facilities for modules installed in the MVS link pack area from the CICS APF-authorized library, *hlq.SDFHAUTH*. That is, such a module is used from the first of the following locations that it is found in:

1. STEPLIB concatenation
2. MVS link pack area
3. MVS LNKLST

#### Using modules from the MVS link pack area

To use any of the CICS modules installed in the MVS link pack area from the *hlq.SDFHAUTH* library you must remove any version of the module from the *hlq.SDFHAUTH* library (or any other library in the STEPLIB concatenation).

#### Using modules from the STEPLIB

You can prevent CICS using modules installed in the MVS link pack area from the *hlq.SDFHAUTH* library by installing versions of those modules in a library in the STEPLIB concatenation. CICS then uses the versions of the modules from the STEPLIB concatenation into the CICS address space, rather than any versions that may be in the MVS link pack area.

### Modules in the MVS link pack area from hlq.SDFHLOAD

The use of CICS modules installed in the MVS link pack area from the *hlq.SDFHLOAD* library is controlled by CICS system initialization parameters and resource definitions.

The *hlq.SDFHLOAD* library is used for non-nucleus CICS modules, and some CICS nucleus modules. You can also use the library for your own user application programs.

#### Using modules from the MVS link pack area

To use any of the CICS modules installed in the MVS link pack area from the *hlq.SDFHLOAD* library:

- Copy the modules into a CICS LPA library. (That is, you do not have to remove them from the *hlq.SDFHLOAD* library.)
- Specify the system initialization parameter LPA=YES. CICS then uses the following search order:
  1. MVS link pack area
  2. DFHRPL DD concatenation
- For a non-nucleus CICS module or user application program, specify USELPACOPY=YES on the associated PROGRAM resource definition. These modules are identified in the CICS-supplied USERMODs by the statement:

```
/* Not loaded from LPA unless USELPACOPY is set to Y in the CSD */
```

For each CICS-supplied LPA-eligible module that needs USELPACOPY(YES) specified in its associated PROGRAM resource definition, you must create your own resource definition with USELPACOPY(YES) specified, and use it instead of the CICS-supplied resource definition. This is because you cannot modify the CICS-supplied resource definitions. For example, you could use the DFHCSDUP utility program to:

1. Copy the CICS-supplied resource groups that contain the module definitions to new resource groups.
2. For each module that needs USELPACOPY(YES), change the PROGRAM resource definition in the new resource groups to specify USELPACOPY(YES).
3. Add your new resource groups to a new group list (that is, at the start of the list).
4. Append the CICS-supplied group list DFHLIST (or your own equivalent of that group list) to your group list. Alternatively, include DFHLIST on the GRPLIST system initialization parameter as well as your group list.
5. Remove the CICS-supplied groups that you have copied.

Once the program definitions have been changed on the CSD you should:

- Reinitialize the CICS catalogs if you have been using modules not in the MVS link pack area, and now want to use those modules from the MVS link pack area
- Specify your new group list (and DFHLIST if your group list does not include the list of resource groups provided in DFHLIST) on the GRPLIST system initialization parameter.

A sample DFHCSDUP job for all CICS LPA-eligible jobs is in Figure 5 on page 140.

**Note:** In the above example, instead of steps 3 and 4, you could use the CEDA transaction to:

- Copy your group list to create a new group list.
- Add the new (USELPACOPY(YES)) groups to the new group list *in the same place as* the original, CICS-supplied, groups.

**Note:**

1. CICS uses eligible modules installed in the MVS link pack area, if:
  - You have **not** specified the name of the module on the CICS system initialization parameter PRVMOD.
  - The module has not been already loaded from the DFHRPL concatenation.
2. If CICS cannot find an eligible module in the MVS link pack area, it loads the private (non-shared) version into the CICS address space from the DFHRPL concatenation, after issuing the message DFHLD0109I to warn you that the module is not in the MVS link pack area. (See page “The module-not-found warning message (DFHLD0109I)” on page 139 for more information about this message.)
3. CICS assumes that the PL/I modules, IBMBPSLA and IBMBPSMA, are installed in the MVS link pack area and issues message DFHLD0109I if it fails to find them there. If you want your PL/I application programs to run with the PL/I shared library facility, you must ensure that the modules IBMBPSLA and IBMBPSMA are installed in the MVS link pack area, or in the *hlq.SDFHLOAD* library (or another library in the CICS DFHRPL library concatenation).

4. Program list tables (PLTs) must be placed in the DFHRPL concatenation. However, before PROGRAM resource definitions for phase one PLTPI programs and PLTSD programs are installed (for example, early in CICS initialization) CICS scans the MVS link pack area for those programs, and issues message DFHLD0109I if it cannot find such a program there.
5. Likewise, before PROGRAM resource definitions for global and task-related user exit programs are installed (for example, early in CICS initialization) CICS scans the MVS link pack area for those programs, and issues message DFHLD0109I if it cannot find such a program there.

## Specifying USELPACOPY(YES)

For every non-nucleus CICS module or user application program that you have moved to the MVS link pack area (that is, have removed from the DFHRPL concatenation), ensure that you have specified USELPACOPY(YES) on the associated PROGRAM resource definition. Otherwise, CICS is not able to find the module, and may fail to start up successfully.

## Using modules from DFHRPL

You can prevent CICS from using modules installed in the MVS link pack area from the *hlq.SDFHLOAD* library by either:

- Specifying NO on the LPA system initialization parameter.

This prevents CICS from using any modules installed into the MVS link pack area from the *hlq.SDFHLOAD* library. CICS tries to load the modules from libraries in the DFHRPL concatenation.

You might use this option when you want to run CICS to test a lot of LPA-eligible modules before installing them in the MVS link pack area. For example, you could add the *hlq.SDFHLPA* library to the DFHRPL concatenation while testing CICS modules for the MVS link pack area. Once you have verified the use of those modules from the MVS link pack area, you should specify the LPA=YES system initialization parameter, and remove the *hlq.SDFHLPA* library from the DFHRPL concatenation.

- Specifying the name of the module on the PRVMOD system initialization parameter:

```
PRVMOD={name|(name1,name2,...)}
```

This prevents CICS from using the specified modules from the MVS link pack area for only the run of CICS on which the PRVMOD parameter is specified. You might use the PRVMOD parameter when you want to run CICS to test a new version of an LPA-eligible module before replacing the version already in the MVS link pack area.

Specify the full module name on the PRVMOD parameter, including any suffix (for example, DFHMCP1\$). If only one module is named, the parentheses are optional. The PRVMOD parameter may span input lines. However, do not split module names across lines, because CICS system initialization adds a comma at the end of every input line that does not already end with a comma. The only validity check performed on a module name is to ensure that it does not exceed eight characters.

You cannot code the PRVMOD parameter in the DFHSIT module; you can specify it in the PARM parameter, in the SYSIN data set, or through the system console.

- For a non-nucleus CICS module or user application program, specifying USELPACOPY(NO), the default, on the associated PROGRAM resource definition. These modules are identified in the CICS-supplied USERMODs by the statement:



```
/* Not loaded from LPA unless USELPACOPY is set to Y in the CSD */
```

You might use the USELPACOPY(NO) option of the PROGRAM resource definition for a more permanent exclusion of an LPA-resident module than for the single run of CICS control provided by the PRVMOD system initialization parameter.

### Verifying modules for the MVS link pack area

While verifying new versions of modules to be installed into the MVS link pack area, you can instruct a CICS region to use the new versions from the DFHRPL concatenation by any of the following options:

- The LPA=NO system initialization parameter
- The PRVMOD system initialization parameter
- The USELPACOPY(NO) option of the associated PROGRAM resource definition (where applicable)

For further information about these options, see “Using modules from DFHRPL” on page 138

In all cases, you must install the new versions of the modules into the *hlq.SDFHLOAD* library, or another library in the DFHRPL concatenation.

If you are verifying many CICS LPA-eligible modules, you might like to add the *hlq.SDFHLPA* library to the DFHRPL concatenation. This allows you to check that the modules you have installed in the MVS link pack area are being loaded from there.

**Note:** The CICS-supplied usermods use SMP/E to move CICS LPA-eligible modules into the *hlq.SDFHLPA* library. Similarly, if you use SMP/E to apply service to any of those modules, the versions in the *hlq.SDFHLPA* library is updated. The updated versions of the modules are used from the MVS link pack area after you next re-IPL your MVS with CLPA specified. Until then, if you add the *hlq.SDFHLPA* library to the DFHRPL concatenation of your CICS region, and specify that CICS is not to use the version of the modules in the MVS link pack area, the updated versions of the modules are used from the DFHRPL concatenation.

After you have installed and verified the use of modules from the MVS link pack area, you should remove the versions of the modules from the DFHRPL concatenation of your CICS startup job.

You can find out whether CICS is loading modules from the MVS link pack area or the DFHRPL concatenation by reviewing the index of a system dump for the CICS region started with the system initialization parameter LPA=YES. Modules loaded from the MVS link pack area have the dump option LD=3.

### The module-not-found warning message (DFHLD0109I)

CICS issues message DFHLD0109I if it searches the MVS link pack area for a module installed there from *hlq.SDFHLOAD* and fails to find it.

If you encounter this message, check that you have specified USELPACOPY(YES) on the associated PROGRAM resource definition (if applicable). For further information about using modules loaded in the MVS link pack area from the *hlq.SDFHLOAD* library, see “Modules in the MVS link pack area from *hlq.SDFHLOAD*” on page 136.

CICS uses console routing code 11 for this particular message, which allows you to control the output of this message. For example, you can:

1. Exclude, as required, routing code 11 from specific MVS console definitions in the CONSOLxx member of SYS1.PARMLIB.

2. Use the MVS VARY command to prevent this message from appearing on specified consoles by omitting route code 11 from a VARY command that defines which routing codes go to specified devices. For example:

```
VARY devnum,CONSOLE,ROUT=(rtcode,rtcode,.,.,.)
```

Alternatively, you can remove route code 11 from those already defined by using the following VARY command:

```
VARY devnum,CONSOLE,DROUT=(11)
```

3. Use the MVS message processing facility (MPF) to inhibit the message. To use this facility, code an entry specifying the CICS message number in the MPFLSTxx member of SYS1.PARMLIB.

CICS assumes that the following PL/1 modules are LPA eligible and issues message DFHLD0109I if it fails to find them there:

- IBMBPSLA
- IBMBPSMA

---

## Sample DFHCSDUP job to specify USELPACOPY(YES)

The standard IBM-supplied program definitions in the CSD all specify USELPACOPY(NO). If you copy, or move, to the LPA some (or all) of the IBM programs defined by definitions in the CSD, the next step is to modify the USELPACOPY attribute to ensure CICS uses the LPA copy.

To simplify this task:

- IBM supplies, in the DFH\$ULPA member of the SDFHSAMP library, an alternate set of DEFINE statements for all the IBM-supplied programs. All the programs defined in DFH\$ULPA specify USELPACOPY(YES).
- If you don't want all the programs to be defined for LPA use, edit the member to remove the programs that are to remain as USELPACOPY(NO).
- The USELPACOPY(YES) versions are all defined in one new group called DFH\$ULPA. Change this group name if you want to use your own name.
- Run the sample DFHCSDUP job shown in Figure 5 to add the DFH\$ULPA versions of the definitions to your CSD.
- There is no need to remove the standard definitions from DFHLIST. Specifying your group list after DFHLIST on the GRPLIST system initialization parameter ensures that the modified definitions override the standard definitions.

```
//LPAMODS    JOB (account_details),MSGCLASS=A,MSGLEVEL=(1,1),
//           CLASS=A,NOTIFY=userid
//DEFULPA    EXEC PGM=DFHCSDUP
//STEP LIB   DD DSN=CICSTS32.CICS.SDFHLOAD,DISP=SHR
//SYS PRINT  DD SYSOUT=*
//DFHCSD     DD DSN=user.CICSTS32.CICS.DFHCSD,DISP=OLD
//SYS IN     DD DSN=CICSTS32.CICS.SDFHSAMP(DFH$ULPA),DISP=SHR
//*
//*
```

Figure 5. Sample DFHCSDUP job for all CICS LPA-eligible modules

---

## Chapter 16. Installing CICSplex SM modules in the MVS link pack area

The benefits of using the MVS link pack area (LPA) are:

- **Sharing** – modules in the LPA can be shared by two or more CICS regions in the same MVS image, giving an overall reduction in the total working set.
- **Integrity** – the LPA is page-protected, even against key 0 programs, so all modules placed there are automatically protected against overwriting by other programs such as CICS applications. (This integrity feature applies equally to a single CICS system within the processor.)

Every CICSplex SM module installed in the LPA can be used only by the release of CICSplex SM to which it relates.

CICSplex SM supplies prebuilt SMP/E USERMODs as members in the CICSTS32.CPSM.SEYUSAMP library. The USERMOD is:

```
EYU$UM01 - Local MAS modules
```

These USERMODs contain ++MOVE statements for each module that is eligible for the extended link pack area (ELPA). A read-only module that may reside above 16MB is eligible for the ELPA.

CICSplex SM allocates an empty library for your use, called SYS1.CICSTS32.CPSM.SEYULPA. You can use SYS1.CICSTS32.CPSM.SEYULPA as the LPA library or you can add the modules to another LPA library.

If you are going to use SYS1.CICSTS32.CPSM.SEYULPA, verify that you have already authorized this library (see Chapter 7, “Authorizing the CICS and CICSplex SM libraries,” on page 79), and that you have applied appropriate security (see the *CICS RACF Security Guide*.) You can give the SYS1.CICSTS32.CPSM.SEYULPA library your own high-level index. If you do, you must specify the new index on the LINDEX parameter of the DFHISTAR job.

The following sections provide information about:

- Space requirements
- Installing CICSplex SM modules into the LPA
- Applying maintenance to LPA modules.

---

### Space requirements

You must allow enough space in the link pack area for the installation of the selected CICSplex SM modules.

The total space needed depends on how the modules are packaged into the link pack area by the operating system and a local MAS requires approximately 2034KB.

---

### Installing CICSplex SM modules into the LPA

The term *install* means move or copy a module into the SYS1.CICSTS32.CPSM.SEYULPA library, by using SMP/E, or by using a copying method that reblocks the copied modules when the target data set has a smaller block size than the data set you are copying from (for example, use the COPYMOD

function of the IEBCOPY program). The procedure for installing modules in the LPA by using SMP/E is described in this section.

CICSplex SM modules that are eligible for inclusion in the link pack area are listed in Appendix B, "CICSplex SM modules eligible for the MVS link pack area," on page 393

CICSplex SM creates default definitions for the LPA-eligible modules specifying USELPACOPY(YES). This means that you do not need to modify these definitions to be able to search the LPA. You can, however, specify whether the LPA is searched by using the CICS system initialization parameters LPA and PRVMOD. Specify LPA=YES if you need CICS to search the LPA for these programs. If you specify LPA=NO, which is also the default, the LPA is not searched for these programs.

If you specify LPA=YES but have not moved the modules to the LPA, message DFHLD0109I is issued for each module. If you specify LPA=YES and have moved these modules to the LPA, you can use the PRVMOD system initialization parameter to control which modules are to be used from the LPA. See "Using modules from DFHRPL" on page 138 for more information.

You should not relink-edit the modules in order to get them into the SYS1.CICSTS32.CPSM.SEYULPA library. CICSplex SM modules, as supplied, have the necessary attributes that cause MVS to load them automatically above 16MB (into the ELPA).

The MVS link pack area has both pageable and fixed areas. Although you can install CICSplex SM modules into the fixed areas, for performance reasons you should use the pageable areas.

Modules to be loaded into the MVS pageable link pack area (PLPA) must have been link-edited with the RENT attribute. The library in which these modules reside must be named in an LPALSTxx member of the SYS1.PARMLIB library.

To install modules in the CICSplex SM LPA library, and to ensure that SMP/E can continue to service them, complete the following steps for one or both of the CICSplex SM-supplied USERMODs:

1. Receive the USERMOD into the CICSplex SM global zone, and apply it to the CICSplex SM target zone.
2. Define the SYS1.CICSTS32.CPSM.SEYULPA library to your MVS.

## Receiving and applying the USERMOD

To receive and apply the CICSplex SM-supplied USERMOD, in EYU\$UM01, you can use the sample job EYULPMOD, which is tailored by DFHISTAR and stored in theCICSTS32.XDFHINST library when you run the DFHISTAR job.

Receive the USERMOD into the CICSplex SM global zone and apply it to the CICSplex SM target zone. This causes SMP/E to move those load modules you have specified from the named CICSplex SM target library (either CICSTS32.CPSM.SEYUAUTH or CICSTS32.CPSM.SEYULOAD) into the SYS1.CICSTS32.CPSM.SEYULPA library.

When the USERMOD is applied, the corresponding LMOD entries within the target zone SMP CSI are updated. Either or both USERMODs may be applied depending on your enterprise's requirements.

Do not accept the USERMOD into the distribution zone, and, for the time being, do not apply it to any other target zone.

## Defining the **SYS1.CICSTS32.CPSM.SEYULPA** library to your MVS

Add the full name of the SYS1.CICSTS32.CPSM.SEYULPA library to an LPA1STxx member of SYS1.PARMLIB. This ensures that the library contents are loaded into the LPA at the next IPL of your system when CLPA is specified.

When you have defined the SYS1.CICSTS32.CPSM.SEYULPA library to MVS, you should re-IPL your MVS with CLPA specified to enable the modules in the SYS1.CICSTS32.CPSM.SEYULPA library to be used from the LPA.

| Use the USERMOD EYU\$UM01, supplied in the SEYUSAMP library, to move the  
| LPA-eligible modules into the SEYULPA library. These modules are listed in  
| Appendix B, "CICSplex SM modules eligible for the MVS link pack area," on page  
| 393.

---

## Applying maintenance to LPA modules

Use the SMP/E RESTORE function to back off the USERMOD before modules in the LPA are updated or copied. Afterwards, the USERMOD may be reapplied.



---

## Chapter 17. Defining CICS IPCS exit control data to MVS

If you use the MVS interactive problem control system (IPCS) to format and analyze CICS system dumps, you should ensure that the release-specific CICS formatting routines are defined and available to MVS.

The formatting routine for use under IPCS has the release identifier as part of its name; that is, DFHPD**650**. This is the formatting routine you must define to IPCS when formatting system dumps. The CICS formatting routine is release-specific, so if you run more than one release of CICS, ensure that you use the correct version for the system dump you are formatting.

---

### The DFHIPCSP CICS exit control data

IPCS provides an exit control table with imbed statements so that other products can supply exit control information.

The IPCS default table, BLSCECT, normally in the SYS1.PARMLIB library, has the following entry for CICS:

```
IMBED MEMBER(DFHIPCSP) ENVIRONMENT(ALL) /* CICS          */
```

Ensure that your IPCS job can find the CICS-supplied DFHIPCSP module. The DFHIPCSP module is in the *h/q*.SDFHPARM library. You can either copy the DFHIPCSP module into SYS1.PARMLIB (so that it is in the same default library as BLSCECT) or provide an IPCSPARM DD statement to specify the library containing the IPCS control tables. For example:

```
//IPCSPARM DD DSN=SYS1.PARMLIB,DISP=SHR          For BLSCECT  
//          DD DSN=CICSTS32.CICS.SDFHPARM,DISP=SHR      For DFHIPCSP
```

The following example code shows the release-specific entries that are specified in DFHIPCSP.

```

/* ===== */
EXIT EP(DFHPD212) VERB(CICS212) ABSTRACT(+
'CICS Version 2 Release 1.2 analysis')
EXIT EP(DFHPD321) VERB(CICS321) ABSTRACT(+
'CICS Version 3 Release 2.1 analysis')
EXIT EP(DFHPD330) VERB(CICS330) ABSTRACT(+
'CICS Version 3 Release 3 analysis')
EXIT EP(DFHPD410) VERB(CICS410) ABSTRACT(+
'CICS Version 4 Release 1 analysis')
EXIT EP(DFHPD510) VERB(CICS510) ABSTRACT(+
'CICS Transaction Server for OS/390 Release 1 analysis')
EXIT EP(DFHPD520) VERB(CICS520) ABSTRACT(+
'CICS Transaction Server for OS/390 Release 2 analysis')
EXIT EP(DFHPD530) VERB(CICS530) ABSTRACT(+
'CICS Transaction Server for OS/390 Release 3 analysis')
EXIT EP(DFHPD610) VERB(CICS610) ABSTRACT(+
'CICS Transaction Server for z/OS V2 R1 analysis')
EXIT EP(DFHPD620) VERB(CICS620) ABSTRACT(+
'CICS Transaction Server for z/OS V2 R2 analysis')
EXIT EP(DFHPD630) VERB(CICS630) ABSTRACT(+
'CICS Transaction Server for z/OS V2 R3 analysis')
EXIT EP(DFHPD640) VERB(CICS640) ABSTRACT(+
'CICS Transaction Server for z/OS V3 R1 analysis')
EXIT EP(DFHPD650) VERB(CICS650) ABSTRACT(+
'CICS Transaction Server for z/OS V3 R2 analysis')
/* ===== */

```

Figure 6. Release-specific entries in DFHIPCSP for DFHPDnnn routines

To use the DFHIPCSP member as it is, rename the CICS-supplied version of DFHPDX for earlier releases to the names that are shown in the table.

For information about using IPCS to format CICS system dumps, see the *CICS Operations and Utilities Guide*.

## Implementing changes

Re-IPL MVS to bring the changes that are described in this chapter into effect before you attempt to run the IVPs.



---

## Chapter 18. Preparing to use the CICSplex SM IPCS tools

The interactive problem control system (IPCS) provides MVS users with an interactive facility for diagnosing software failures. You can use IPCS to format and analyze SDUMPs produced by CICSplex SM or stand-alone dumps obtained while CICSplex SM was active in the system being dumped. You can either view the dumps at your terminal or print them.

CICSplex SM provides two types of IPCS tools:

- A set of panels (driven by a corresponding set of CLISTs) that allow you to display:
  - Subsystem information
  - Address space-related control blocks
  - Modules loaded by CICSplex SM
  - Tasks created by CICSplex SM
  - Storage subpools managed by CICSplex SM
  - BBC LU 6.2 communication information
- CICSplex SM provides a dump formatting routine that can be used with the VERBEXIT subcommand to format CMAS or MAS dumps.

For more information about:

- IPCS, see the *MVS Interactive Problem Control System: User's Guide*.
- Using IPCS to format CICSplex SM system dumps, see the *CICS Operations and Utilities Guide*.
- Displaying and formatting dumps with IPCS, see the *CICSplex System Manager Problem Determination* guide.

Before you can use the CICSplex SM IPCS tools, you must make the preparations described in:

- “Updating BLSCECT”
- “Updating library allocations” on page 148
- “SDUMP options” on page 148.

---

### Updating BLSCECT

IPCS provides an exit control table called BLSCECT; it normally resides in SYS1.PARMLIB. This table contains imbed statements to enable other products to supply exit control information. You must perform the following steps:

1. Update the BLSCECT table with the following IMBED statement:

```
IMBED MEMBER(EYUIPCSP) ENVIRONMENT(ALL)
EYUIPCSP identifies the CICSplex SM formatting routine as EYU9D320 with a
VERB name of CPSM320.
```
2. Make sure EYUIPCSP can be found by your IPCS job by doing one of the following:
  - Copy EYUIPCSP from the CICSTS32.CPSM.SEYUPARM library into the same library as BLSCECT (usually SYS1.PARMLIB).
  - Provide an IPCSPARM DD statement to specify the library that contains the IPCS control tables. For example, the DD statement for a batch TSO session might look like this:

```

| //IPCSPARAM DD DSN=SYS1.PARMLIB,DISP=SHR          for BLSCECT
| //          DD DSN=CICSTS32.CPSM.SEYUPARM,DISP=SHR for EYUIPCSP
|

```

For more information about SYS1.PARMLIB library members related to IPCS, see the *MVS Interactive Problem Control System (IPCS): Customization* manual.

---

## Updating library allocations

To update the library allocations, you must do the following:

- Update the CLIST or REXX EXEC that invokes IPCS at your enterprise to include the following data set allocations:

### ISPPLIB

CICSTS32.CPSM.SEYUPLIB

Contains panels that allow you to view data structures.

### SYSPROC

CICSTS32.CPSM.SEYUCLIB

Contains CLISTs that obtain information from a dump and display it. These CLISTs also create a set of IPCS symbol equates to help you locate data while browsing a dump outside of the panels.

### ISPMLIB

CICSTS32.CPSM.SEYUMLIB

Contains messages issued by the CLISTs.

- Make sure that the EYU9D310 IPCS user exit routine is in a library in the linklist or a library that is accessed by the JOBLIB, STEPLIB, or TASKLIB option of the IPCS command, during IPCS session. To accomplish this, do one of the following:
  - Allocate CICSTS32.CPSM.SEYULINK to the desired DD statement.
  - Copy CICSTS32.CPSM.SEYULINK (EYU9D310) to an appropriate library.
  - Invoke IPCS, using the TASKLIB keyword to allocate CICSTS32.CPSM.SEYULINK .

For example, issue the TSO COMMAND:

```
IPCS NOPARM TASKLIB('CICSTS32.CPSM.SEYULINK')
```

---

## SDUMP options

Make sure the following SDUMP options are in effect at the time the dump is taken:

- CSA** Common service area
- LPA** Link pack area modules
- LSQA** Local system queue area
- NUC** Non-page-protected areas of the DAT-on nucleus
- PSA** Prefixed storage area for all processors
- RGN** Private area of address space being dumped
- SQA** System queue area
- SUM** Summary dump
- SWA** Scheduler work area
- TRT** GTF, system trace, and master trace data

---

## Chapter 19. MVS Program properties table entries

There are some CICS properties that you can optionally define to MVS. These are in the following section.

You can define entries for CICS in the MVS program properties table (PPT). Figure 7 is an example of a CICS PPT entry in the SCHEDxx member of SYS1.PARMLIB.

```
/******  
/*          Program Properties table addition          */  
/*          for the CICS program, DFHSIP             */  
/*          */  
/* The following defaults apply to this CICS entry:  */  
/*          */  
/* No affinity to a particular processor      (AFF(NONE)) */  
/* Can be canceled                          (CANCEL)    */  
/* Requires data set integrity              (DSI)        */  
/* Not a privileged job                     (NOPRIV)     */  
/* Password protection is required         (PASS)       */  
/* Not a system task                       (NOSYST)     */  
/* Protection key 8                        (KEY(8))     */  
PPT PGMNAME(DFHSIP) /* Add program name DFHSIP to the PPT*/  
   NOSWAP          /* Non-swappable */  
   NOPREF         /* No preferred storage required */  
/*          */
```

Figure 7. Sample CICS PPT entry

For information about defining options in the PPT, see the *z/OS MVS Initialization and Tuning Guide*.

---

### RACF password checking

If your installation has a PPT entry for the DFHSIP program, you should ensure that the PPTNOPAS option is **not** set in the PPT because this bypasses password and RACF authorization checking. However, you should consider making your CICS regions non-swappable by specifying the PPTNSWP option in the PPT. For information about defining CICS PPT entries in the SCHEDxx member of the SYS1.PARMLIB library, see the *z/OS MVS Initialization and Tuning Reference manual*.

---

### Non-swappable CICS regions

For performance reasons, you should consider making your CICS regions non-swappable, by specifying the NOSWAP option in the PPT. However, you should be aware that the use of certain functions causes CICS regions to be made non-swappable automatically, regardless of what is specified in the PPT (for example, regions using cross-memory services for MRO).

---

### MVS protection key for CICS

CICS can run only in MVS protection key 8 (the default). You must not define any other protection keys for CICS.

If you want to use the storage protection facility of CICS, you must specify the system initialization parameter STGPROT=YES, and must have the required hardware and software. If you operate CICS with storage protection, CICS observes

the storage keys and execution keys that you specify in various system and resource definitions. For information about the CICS storage protection facility, and how it affects the storage allocation for the dynamic storage areas, see the *CICS System Definition Guide*. For information about hardware and software that is required by the CICS storage protection facility, see the *CICS Transaction Server for z/OS Program Directory*.

---

## Chapter 20. MVS performance definitions

You can use the MVS workload management facility to manage sysplex resources across MVS subsystems, in parallel with the existing system resource management facilities.

For information about MVS workload management, see the *z/OS MVS Planning: Workload Management* manual.

If you want to use the MVS workload manager facility, you should:

1. Implement workload management on the MVS images that the CICS workload is to run on, as outlined in “Implementing MVS workload management.”
2. Ensure that CICS performance parameters correspond to the policies defined for MVS workload management, as outlined in “Matching CICS performance parameters to service policies” on page 152.

---

### Implementing MVS workload management

The task of implementing MVS workload management is part of the overall task of planning for, and installing, MVS.

Implementing MVS workload management generally involves the following steps:

1. Establishing your workloads.
2. Setting your business priorities.
3. Understanding your performance objectives.
4. Defining critical work.
5. Defining performance objectives based on current business needs
6. Get agreement for your workload performance objectives.
7. Specify a service level agreement or performance objectives.
8. Specify an MVS WLM service definition that uses the information from step 7.

**Note:** It is helpful at this stage to record your service definition in a form that helps you to enter it into the MVS workload manager ISPF application. You are recommended to use the worksheets provided in the *z/OS MVS Planning: Workload Management* manual, SA22-7602.

9. Install MVS.
10. Set up a sysplex with a single MVS image, and run in workload manager compatibility mode.
11. Upgrade your existing XCF couple data set.
12. Start the MVS workload manager ISPF application, and use it in the following steps.
13. Allocate and format a new couple data set for workload management. (You can do this from the ISPF application.)
14. Define your service definition.
15. Install your service definition on the couple data set for workload management.
16. Activate a service policy.
17. Switch the MVS image into goal mode.
18. Start up a new MVS image in the sysplex. (That is, attach the new MVS image to the couple data set for workload management, and link it to the service policy.)

19. Switch the new MVS image into goal mode.
20. Repeat steps 18 on page 151 and 19 for each new MVS image in the sysplex.

**Note:**

1. Current release support for MVS workload manager is initialized automatically during CICS startup.
2. All CICS regions (and other MVS subsystems) running on an MVS image with MVS workload management are subject to the effects of workload manager.

---

## Matching CICS performance parameters to service policies

You must ensure that the CICS performance parameters are compatible with the workload manager service policies used for the CICS workload.

In general, you should define CICS performance objectives to the MVS workload manager first, and observe the effect on CICS performance. Once the MVS workload manager definitions are working correctly, you can then consider tuning the CICS parameters to further enhance CICS performance. However, you should use CICS performance parameters as little as possible.

Performance attributes that you might consider using are:

- Transaction priority, passed on dynamic transaction routing. (Use prioritization carefully, if at all.) The priority assigned by the CICS dispatcher must be compatible with the task priority that is defined to MVS workload manager.
- Maximum number of concurrent user tasks for the CICS region.
- Maximum number of concurrent tasks in each transaction class.

---

## Chapter 21. MVS automatic restart management definitions

You can exploit the MVS automatic restart management facility that is provided by MVS to implement a sysplex-wide integrated automatic restart mechanism.

If you want to use the MVS automatic restart manager facility, you should:

1. Implement automatic restart management on the MVS images that the CICS workload is to run on.
2. Ensure that CICS startup JCL used to restart CICS regions is suitable for MVS automatic restart management.
3. Specify appropriate CICS START options.
4. Specify appropriate MVS workload policies.

If you do not wish to use the MVS automatic restart management facility, you can use XRF to provide restart of failed CICS regions. For information about XRF, see the *CICS/ESA 3.3 XRF Guide*.

---

### Implementing MVS automatic restart management

The task of implementing MVS automatic restart management is part of the overall task of planning for and installing MVS. For information about MVS automatic restart management, see the *z/OS MVS Setting Up a Sysplex* manual.

Implementing MVS automatic restart management for CICS generally involves the following steps:

- Ensure that the MVS images available for automatic restarts have access to the databases, logs, and program libraries required for the workload.
- Identify those CICS regions for which you want to use automatic restart management.
- Define restart processes for the candidate CICS regions.
- Define ARM policies for the candidate CICS regions.
- Ensure that the system initialization parameter XRF=NO is specified for CICS startup.

For further information on implementing automatic restart management, see the *CICS Recovery and Restart Guide*.





---

## Chapter 22. MVS cross-system MRO definitions

You can use the CICS interregion communication (IRC) facility for multiregion operation (MRO) between CICS regions across MVS images in a sysplex. This exploits the cross-system coupling facility (XCF) of MVS, and makes it unnecessary to use VTAM to communicate between MVS images within the same sysplex.

Within a sysplex, DFHIRP must be installed from the highest release of CICS running in that MVS image.

---

### Sysplex overview

A sysplex consists of multiple MVS systems, coupled together by hardware elements and software services. In a sysplex, MVS provides a platform of basic multisystem services that multisystem applications like CICS can exploit. As an installation's workload grows, additional MVS systems can be added to the sysplex to enable the installation to meet the needs of the greater workload.

To use XCF to communicate in a sysplex, each CICS region joins an XCF group called DFHIR000 by invoking the MVS IXCJOIN macro using services that are provided by the DFHIRP module. The member name for each CICS region is always the CICS APPLID (NETNAME on the CONNECTION resource definition) used for MRO partners. Each CICS APPLID must be unique within any sysplex regardless of the MVS levels that are involved. Within the sysplex, CICS regions can communicate only with members of the CICS XCF group (DFHIR000).

---

### MVS XCF considerations for MRO

Ensure that when you format the primary and alternate couple data sets used by the XCF component of MVS:

- The value specified for the MAXMEMBER parameter is large enough to handle the number of CICS regions and users of the EXCI in the CICS XCF group.

Each XCF group is limited to 1023 members, which is therefore the theoretical maximum number of CICS regions that can participate in XCF/MRO in a single sysplex. However, the maximum size of the XCF group is reduced if you set the MVS MAXMEMBER parameter, used to define XCF couple data sets, to a lower limit. When calculating the maximum number of members in the CICS XCF group, allow one member for:

- Each CICS region to run on an MVS image in the sysplex.
- Each pipe that is allocated by a user of the external CICS interface (EXCI). For information about EXCI users and pipes, see the *CICS External Interfaces Guide*.

To list the members in the CICS XCF group, you can use the MVS DISPLAY command. The name of the CICS group is always DFHIR000, so you could use the MVS command:

```
DISPLAY XCF,GROUP,DFHIR000,ALL
```

- The value specified for the MAXGROUP parameter is large enough for the CICS XCF group to be established.



---

## Chapter 23. PR/SM policy for handling MVS failures

If you are running CICS under MVS in a Processor Resource/Systems Manager™ (PR/SM™) environment, you should define to MVS the preferred XCF PR/SM policy for handling MVS failures in a PR/SM environment, and define to PR/SM the authorization for each LPAR to cause reset or deactivation of another LPAR.

---

### XCF PR/SM policy

The function that enables MVS images to take over the resources of other MVS images in the same sysplex. This function is also known as the PR/SM automatic reconfiguration facility (ARF).



---

## Chapter 24. MVS ASREXIT - SYMREC Authorization Exit

A CICS program may call the first failure symptoms (FFS) component. This uses the MVS SYMREC macro to write symptom records to the MVS SYS1.LOGREC data set, in addition to, or instead of, a job log.

The SYMREC authorization exit, ASREXIT, must be in effect to allow CICS to use the SYMREC macro call, otherwise the call fails with return code 12, reason code 3868 (X'F1C').

When SYMREC is called by CICS, the ASREXIT routine issues a return code that permits the SYMREC to be successfully written.

The MVS sample exit programs ASREXT0 and ASREXT1, supplied in SYS1.SAMPLIB, are suitable for this purpose. For further information about these exits, see the *z/OS MVS Installation Exits* manual. The ASREXIT routine can determine if CICS is the caller by testing EPLPNAME for the value 'DFHSIP' except:

- When DFHSIP is renamed, in which case EPLPNAME contains the new name.
- When DFHSIP is the subject of an MVS LINK, in which case EPLPNAME contains the name of the program issuing the MVS LINK (unless it too is the subject of an MVS LINK).

If you choose this method, you may wish to code your ASREXIT routine to allow for these exceptions.

An alternative method of coding the ASREXIT routine is in Figure 8 on page 160. This method is not affected by the exceptions that are mentioned above.

```

TITLE 'SYMREC SAMPLE EXIT'
ASREPL
PRINT NOGEN
IHAPSA
IKJTCB
PRINT GEN
DFHAFCD
EJECT
ASREXIT CSECT
ASREXIT AMODE 31
ASREXIT RMODE ANY
USING *,R15          Temporary addressability
MODID BR=YES
DROP R15
STM R14,R12,12(R13)  Save the caller's registers
LR R12,R15
USING ASREXIT,R12
L R3,0(,R1)          Load the address of the EPL
USING EPL,R3          Get addressability
LA R15,RCREJECT      Preset "reject" return code
USING PSA,0
L R1,PSATOLD          Point at current TCB
USING TCB,R1
L R1,TCBEXT2          Point at TCB extension
DROP R1
USING TCBXTNT2,R1
ICM R1,B'1111',TCBCAUF Point at AFCB; is there one?
BZ SETRC              No, branch
DROP R1
USING DFHAFCB,R1
CLC AFIDENT,=C'AFCX' Is it a genuine CICS AFCB?
BNE SETRC              No, branch
CLI AFVER,AFVER1      Is it at least Version 1?
BL SETRC              No, branch
AH R1,AFLENG          Add length of AFCB's DW
DROP R1               table.
USING AFTSTART-AFPFXLEN,R1 Allow for AFCB prefix length
ICM R1,B'1111',AFTAFCS Point at AFCS; is there one?
BZ SETRC              No, branch
DROP R1
LA R15,RCWRITE        Set "write" return code
SETRC DS 0H
ST R15,EPLRETC        Store return code
DROP R0
DROP R3
DROP R12
EXIT LM R14,R12,12(R13) Restore caller's registers
BR R14                Return
LTORG *
R1 EQU 1               Register 1
R3 EQU 3               Register 3
R12 EQU 12             Register 12
R13 EQU 13             Register 13
R14 EQU 14             Register 14
R15 EQU 15             Register 15
RCREJECT EQU X'0C'     Return code C
RCWRITE EQU X'00'      Return code 0
END* CONSTANTS

```

Figure 8. An example of coding the ASREXIT routine.

---

## Chapter 25. Definitions required for VSAM RLS support

If you plan to use VSAM RLS to enable CICS regions to share VSAM data sets, carry out the following steps:

1. Define the master coupling facility lock structure.
2. Define coupling facility cache structures and cache sets.
3. Define SMS storage classes for RLS access.
4. Alter data set characteristics, if necessary, to make data sets eligible for RLS access.
5. Define sharing control data sets.
6. Establish new authorization that is required by the VSAM RLS support.
7. Add new parameters to SYS1.PARMLIB.
8. Establish new procedures for VSAM RLS support.
9. Activate the coupling facility structures.

An overview of each of these steps follows.

---

### Defining the master coupling facility lock structure

VSAM RLS support requires the coupling facility to define a master lock structure, IGWLOCK00, for cross system locking.

See the *z/OS DFSMSdfp Storage Administration Reference* manual for information about calculating the size you need for the lock structure.

The amount of coupling facility space required depends on several characteristics of your hardware configuration and the applications that you run, such as:

- The number of processors you have
- The power of your processors
- Your ratio of non-update activity to update activity
- Your ratio of recoverable updates to non-recoverable updates
- Your ratio of sequential requests to direct requests

You define the lock structure in the CFRM policy with the IXCMIAPU utility.

---

### Defining coupling facility cache structures and cache sets

VSAM RLS support requires the coupling facility to define **cache structures** for cross system buffer invalidation. You need to determine the number and size of cache structures you require.

The number needed depends on factors such as:

- The number of coupling facilities you have
- The amount of space in each coupling facility
- The amount of data that is accessed through each coupling facility

See the *z/OS DFSMSdfp Storage Administration Reference* manual for information about calculating the amount of space you will need for the cache structures. If you have previously used data sets in LSR mode, the total amount of coupling facility

space allocated to cache structures should not be less than the amount of storage you were using for LSR pools, including hiperspace buffers (if used).

You can achieve performance benefits by:

- Making the size of the cache larger
- The way in which you divide cache structures across coupling facilities

You define cache structures in the CFRM policy with the IXCMIAPU utility.

## Defining cache sets

You define cache sets with the ISMF control data set (CDS) application.

A cache set maps on to one or more cache structures. If more than one cache set is specified, the data sets can be re-bound to another cache structure in the set in the event of a cache structure failure.

See the *z/OS DFSMSdfp Storage Administration Reference* manual for more information about cache sets.

---

## Defining SMS storage classes for RLS access

Before you can use VSAM RLS, you need one or more storage classes which specify a non-blank cache set name.

The ISMF storage class application allows you to specify a cache set name when defining or altering a storage class, together with weighting parameters for tuning, such as CF DIRECT WEIGHT and CF SEQUENTIAL WEIGHT. See the *z/OS DFSMSdfp Storage Administration Reference* manual for more information about defining SMS storage classes.

---

## Altering data set attributes for RLS access

Before you can use a data set in RLS access mode, you must ensure that it is eligible. To be eligible for RLS:

- Data sets must reside in SMS managed storage.
- Data sets must specify a storage class that has a non-blank cache set name.
- Data set recoverability attributes must be defined in the ICF catalog (not in the CICS file control resource definition, where they are ignored for RLS).

You can specify a data set's attributes using the Access Method Services (AMS) DEFINE CLUSTER or ALTER CLUSTER commands.

Specifying a LOG parameter of NONE, UNDO or ALL ensures that the recoverability of the data set is not undefined. You cannot open files in RLS mode if the LOG parameter of the associated data set is UNDEFINED. If you specify LOG(ALL), you must also specify a forward recovery log stream on the LOGSTREAMID parameter.

To use backup while open (BWO) for an RLS-accessed sphere, specify the BWO parameter. Specifying BWO(TYPECICS) means that backup while open can be used. All other values for BWO (including undefined) mean backup while open is not allowed. BWO(TYPECICS) is only valid if LOG(ALL) and LOGSTREAMID are also specified.

- Data sets must not specify the IMBED attribute.



If you have some data sets that specify imbed, you must remove the IMBED option before you can use the data sets in RLS mode. Redefine a new data set without IMBED and use the AMS REPRO function to copy the old data set to the new data set.

**Note:** RLS supports the REPLICATE cluster attribute. It does not provide any performance benefit, and removing it could save DASD space.

---

## Defining sharing control data sets

VSAM RLS requires sharing control data sets. These are used to maintain data integrity in the sharing environment. The sharing control data set is used sysplex-wide by all the SMSVSAM servers, and is always duplexed.

Two active (and at least one spare) sharing control data sets must be available at all times.

The size required depends on the number of MVS images in the sysplex, and on the number of files that are expected to be open concurrently. The *z/OS DFSMSdfp Storage Administration Reference* manual gives information about calculating the amount of space that is needed for the sharing control data sets.

Sharing control data sets are VSAM linear data sets that must reside on volumes which have global connectivity. The data sets may have multiple extents, but only on the same volume. You define them using standard techniques for defining data sets. The names must have SYS1.DFPSHCDS as the first and second qualifiers. See the *z/OS DFSMSdfp Storage Administration Reference* manual for other rules relating to the definition of sharing control data sets.

You must not issue RESERVEs on any volumes on which sharing control data sets reside. Convert any such RESERVEs to enqueuees.

You can check that the data sets are available to the sysplex with the MVS DISPLAY SMS command, on any MVS image, as follows:

```
D SMS,SHCDS
```

This command shows the names of the two active, and the spare data set as in the following example:

Name	Size	%UTIL	Status	Type
ACTIVE1.VP2SS03 7920KB	7920KB	74%	Good	ACTIVE
ACTIVE2.VP2SS03 7920KB	7920KB	74%	Good	ACTIVE
SPARE.VP2SS03 7920KB	7920KB	74%	Good	SPARE

**Note:** The DISPLAY command shows only the third and fourth qualifiers of the sharing control data set names; the first and second qualifiers are always SYS1.DFPSHCDS.

The first time an SMSVSAM server is started in the sysplex, the sharing control data sets need to be varied online using

```
V SMS,SHCDS,NEW
```

for the active data sets, and

```
V SMS,SHCDS,NEWSPARE
```

for the spare data set (or data sets). The server cannot come up properly if this is not done.

---

## Authorizing CICS userids for VSAM RLS support

Authorize each CICS userid that is to use VSAM RLS support to have read access to a profile in the SUBSYSNUM class which matches the applid. See “Authorizing access to an SMSVSAM server” on page 83 for more information.

You may want to restrict access to the AMS SHCDS LIST and REMOVE commands. The *DFSMS/MVS Access Method Services for ICF* manual gives information about using these commands.

---

## Adding new parameters to SYS1.PARMLIB(IGDSMSxx)

To include RLS support in your system, specify the required parameters in the IGDSMSxx member of SYS1.PARMLIB, as follows:

- Specify RLSINIT(YES), otherwise SMSVSAM will not initialize automatically when you IPL MVS. Alternatively, you can start SMSVSAM using the VARY SMS,SMSVSAM,ACTIVE command.
- Specify a value for the deadlock detection interval with the DEADLOCK\_DETECTION parameter.
- Specify time intervals for the creation and synchronization of VSAM RLS SMF records with the CF\_TIME and SMF\_TIME parameters.
- Specify the maximum size of the SMSVSAM local buffer pool with the RLS\_MAX\_POOL\_SIZE parameter.

See the *z/OS DFSMSdftp Storage Administration Reference* manual for information about these parameters.

---

## Establishing new procedures for VSAM RLS support

New operational procedures may be needed in a number of areas as a result of using VSAM RLS support. Areas to consider include:

- **Integrity of data in coupling facility caches**

To make sure that non-IBM products or user programs do not compromise the integrity of data in coupling facility caches when they modify the data on a volume, you should either vary the volume offline to each system in the sysplex, or CF-quiesce the volume using the

```
V SMS,CFVOL(volId),QUIESCE
```

command before running such programs

- **Management of the coupling facility and coupling facility structures**
- **Use of RESERVEs on volumes which contain sharing control data sets**

Make sure that this does not happen. You should convert RESERVEs on other volumes into enqueues.

- **Switching to non-RLS mode in order to run batch update jobs against recoverable data sets**

This subject is in greater detail in the *CICS Recovery and Restart Guide*.

- **Management of forward recovery and your forward recovery logs**

The differences from forward recovery for non-RLS access are:

- The forward recovery log stream must be in the ICF catalog.

- All forward recovery log records for a data set merge into the same log stream.
- Your forward recovery procedure needs to use the SHCDS FRSETRR, FRUNBIND, FRBIND, and FRRESETRR commands (CICSVR version 2 release 3 automatically issues these commands).

Refer to the DFSMS/MVS Version 1 Release 3 documentation for more details.

---

## **Activating the coupling facility structures**

Once defined in the CFRM policy, the coupling facility structures must be activated using the SETXCF START POLICY command, specifying a TYPE of CFRM and the policy name.



---

## Chapter 26. Console messages

The message domain supports the use of MVS message routing codes in the range 1 to 16 for those messages that are sent to the console. By default, if the issuing module specifies only CONSOLE (without a qualifying number) as the destination, CICS routes the message with MVS route codes 2 and 11 (for the master console – information). This support is available for all domain-type messages of the form DFHxxnnnn, where xx is the domain code, and nnnn is the message number.

CICS issues other messages (of the form DFHnnnn) with either no route code, or route codes other than 2 and 11.

The physical destination of these messages is controlled by the ROUTECODE parameter on the MVS console entries in a SYS1.PARMLIB member, CONSOLEnn. For further information about MVS console definitions, see the *z/OS MVS Initialization and Tuning Guide* .



---

## Chapter 27. Defining the logger environment for CICS journaling

CICS uses the MVS system logger for all its logging and journaling requirements. Using services provided by the MVS system logger, the CICS log manager supports:

- The CICS system log, which is used for:
  - Dynamic transaction backout
  - Warm and emergency restarts
  - Cold starts, but only if the log contains information required for resynchronizing in-doubt units-of-work
- Forward recovery logs, auto-journals, and user journals.

The MVS system logger is a component of MVS. It provides a programming interface to access records on a log stream. For information about the MVS system logger, see the following MVS publications:

- *z/OS MVS Setting Up a Sysplex* for:
  - General information about the MVS system logger
  - Information about defining and formatting the LOGR couple data set
  - Information about how to plan the system logger configuration, plan and set up a system logger application, and plan for recovery for system logger applications.
- *z/OS MVS Programming: Assembler Services Reference, Volume 1* and *z/OS MVS Programming: Assembler Services Reference, Volume 2* for the syntax of system logger services
- *z/OS MVS Initialization and Tuning Reference* for information about the COUPLExx PARMLIB member.

---

### Requirements planning and checklist

This section summarizes the requirements, and the steps that you need to follow, to set up the CICS logging environment.

Some of the steps listed have a pointer to sections that provide more detailed information. These steps cover both MVS and CICS system programmer tasks and some security administrator tasks, and close cooperation between all groups is needed:

#### Planning

Consider the possible storage options, and choose which of the 3 available hardware options you want to use:

- **Non-volatile coupling facility**, where log stream data is duplexed in the MVS logger data space. Non-volatile storage involves the use of battery backup or an uninterruptible power supply (UPS):
  - When using a UPS, you use a hardware console command to update coupling facility status
  - When using battery backup, batteries must be online and charged.
- **Volatile coupling facility**, where log stream data is duplexed to a staging data set.
- **DASD-only**, where log stream data is duplexed in the MVS logger data space.

See “Coupling facility or DASD-only?” on page 176, to help you decide on one of these or a combination of both.

As part of the planning phase:

- Determine the number CICS regions that require logger support and hence system log streams.
- Determine the number of user journals and autojournals that your regions use.
- Determine the number of forward recovery logs required for VSAM data sets.
- Determine whether any user journal or forward recovery log streams are to be shared between regions (to create merged data automatically).

**Note:** The system log streams, DFHLOG and DFHSHUNT, cannot be shared.

DASD-only log streams can be shared only within the same MVS image.

- Decide on the number and sizes of the coupling facilities to be used.
- Determine the log stream sizes:
  - For coupling facility log streams, see “Coupling facility log streams” on page 177.
  - For DASD-only log streams, see “DASD-only log streams” on page 195.

For information about types of coupling facility, see the *z/OS MVS Setting Up a Sysplex*. Note that the minimum level of coupling facility supported by the MVS system logger is CFLEVEL=1, with the appropriate service level of the coupling facility control code that supports CFLEVEL=1.

#### **Maintenance**

Ensure that all maintenance affecting the MVS system logger, and the CICS log manager and its utilities, is applied:

- Logger serviceability APARs relating to the MVS system logger are identified with the LOGRSERVICE keyword.
- APARs relating to the CICS log manager are identified with the CICSLOGR keyword.

#### **Create and format the LOGR couple data sets**

In consultation with your MVS system programmer:

- Use MVS utility IXCL1DSU to create and format the primary and alternate LOGR couple data sets.
- Identify the LOGR couple data sets to the sysplex in the COUPLExx member in SYS1.PARMLIB
- Make the LOGR couple data set available to the sysplex.

See “Format the LOGR Couple Data Set and Make it Available to the Sysplex” in the *z/OS MVS Setting Up a Sysplex* manual for information about these steps.

#### **Note:**

1. For this task you need know the number of log streams and, for coupling facility log streams, the number of structures. Each CICS region needs two system log streams and, optionally:
  - A log stream for the log of logs
  - One or more log streams for forward recovery logs



- One or more log streams for autojournals
  - One or more log streams for user journals
2. If you are migrating from a release of OS/390, before release 3, (to meet the z/OS minimum requirement level for CICS TS) , you must reformat your LOGR data sets, because logging enhancements introduced in OS/390 Release 3 required LOGR couple data set of the sysplex to be formatted using OS/390 Release 3 or later.

The removal of the 168 data set limit (described in “General logs” on page 203) also requires the LOGR data set to be formatted with DSEXTENT(nnnnn). This does not apply to LOGR couple data sets formatted for DASD-only logging.

#### **Define coupling facility structures**

If you are using the coupling facility for some or all of your log streams, update your CFRM policy and your LOGR couple data set with the required structure definitions.

See “Defining coupling facility structures” on page 178 for details, including a sample job.

#### **Establish the required security authorizations**

Ensure that all the userids that are involved with running the system logger, or defining or accessing logger resources, are authorized, and that the required profiles are defined in the LOGSTRM general resource class :

- If the MVS system logger address space (IXGLOGR) is not given SAF privileged or trusted status, ensure you give the required authorization to the userid that runs IXGLOGR. For example, if the userid that runs IXGLOGR (defined in the RACF started procedures table (ICHRIN03), or defined in the RACF STARTED class profile) is SYSTASK:
  - SYSTASK requires ALTER access to IXLSTR structure profiles in the FACILITY general resource class for access to log stream coupling facility structures.
  - SYSTASK requires ALTER access to the data set profiles (*hlq.data\_set\_name*) in the DATASET general resource class, for each DASD log stream and staging data set.
- To use the MVS system logger IXCMIAPU utility to define, update and delete entries in the LOGR couple data set, you need appropriate authorizations to the relevant RACF profiles in the LOGSTRM and FACILITY general resource classes. See “Authorizations for users of IXCMIAPU” on page 84 for information and examples of how to do this.
- To enable CICS to create log streams dynamically, and to write to log streams, ensure that the CICS region userid has the required authorizations. See “Authorizations for CICS regions” on page 85 for information and examples of how to do this.

For more information about authorizations for the system logger, see the *z/OS MVS Setting Up a Sysplex* manual

#### **Check sysplex definition in PARMLIB**

To use the MVS system logger, each MVS image must be a member of a sysplex. Ensure your sysplex definition, in PARMLIB member IEASYSxx, specifies either PLEXCFG(MONOPLEX), for a single-member sysplex, or PLEXCFG(MULTISYSTEM), for a multi-member sysplex. Also ensure that you define a COUPLExx member in PARMLIB.

**Note:** The value specified on the SYSPLEX parameter in COUPLExx forms part of DASD-only and staging data set names.

#### **Activate the LOGR subsystem**

Ensure the LOGR subsystem is active to enable the CICS log manager batch utility, DFHJUP, to format and print log data. The LOGR subsystem is defined by the following entry in IEFSSNxx PARMLIB member:

```
SUBSYS SUBNAME(LOGR) INITRTN(IXGSSINT)
```

#### **Plan staging data set requirements**

Staging data sets are used for both DASD-only and coupling facility log streams, and if specified are dynamically allocated by the MVS system logger:

- For DASD-only log streams, staging data sets are the primary (interim) storage.
- For coupling facility log streams, staging data sets are allocated by the system logger to safeguard log data in the event of the log data being in a volatile configuration; that is:
  - There is a loss of the coupling facility battery backup
  - A structure failure that results in the only copy of log data being in MVS local storage buffers.

Consider the following parameters:

- STG\_DUPLEX(YES) and DUPLEXMODE(COND) to cause the system logger to use staging data sets if the coupling facility is not failure independent (see “Staging data sets for coupling facility log streams” on page 194 for more information)
- STG\_MGMTCLAS to specify the System Managed Storage (SMS) management class to be used for staging data set allocation (valid only when STG\_DUPLEX(YES) or DASDONLY(YES) is specified)
- STG\_STORCLAS to specify the SMS storage class to be used for staging data set allocation (valid only when STG\_DUPLEX(YES) or DASDONLY(YES) is specified)
- STG\_SIZE to specify the size of staging data sets
- SHAREOPTIONS(3,3) for log stream data sets and staging data sets (see “VSAM Share Options for System Logger” in *z/OS MVS Setting Up a Sysplex*)

#### **Plan DASD space and SMS environment for logger secondary storage**

System logger secondary storage comprises all log stream (offload) data sets. See “Managing secondary storage” on page 202 for information about size parameters and other attributes relating to secondary storage

#### **Define log streams and log stream models**

Define the specific log streams, and log stream models for dynamic creation of log streams, in the LOGR policy.

In particular, consider the following when defining your log streams:

- Set HIGHOFFLOAD no higher than 80% to allow the offload function to be activated before structures reach the 90% level and provide a buffer to enable CICS to continue writing records without filling the logstream before offload completes.
- Set LOWOFFLOAD for DFHLOG and DFHSHUNT in the range 40–60%. For user journals and the log of logs, specify LOWOFFLOAD as 0.
- Specify HLQ for the high level qualifier for offload data sets—it is not part of the CICS log stream name. The default is IXGLOGR.

- Specify STG\_DUPLEX(YES) and DUPLEXMODE(COND) for log streams in the coupling facility to ensure that staging data sets are used automatically if the coupling facility is volatile or failure dependent.
- Set STG\_SIZE to control the size, in 4K blocks, of staging data sets allocated by the system logger. For coupling facility log streams, the staging data set must hold at least as much data as the log stream in the structure, so that offloads are not triggered by the staging data sets. See “Sizing considerations for coupling facility log streams” on page 184 and “Sizing considerations for DASD-only log streams” on page 197.
- Specify LS\_DATACLAS and LS\_SIZE, for the SMS data class and the number of 4K allocation blocks respectively for log stream off load data sets (see “Managing log data sets” on page 202)
- Specify MODEL(YES) to indicate that a log stream definition is a model only and not an actual log stream. See SDFHINST members DFHILG2 (coupling facility) and DFHILG5 (DASD-only) for samples of model log streams.

**Note:** Use AUTODELETE(YES) with a suitable retention period (RETPD) for general logs but *not* for CICS system logs (DFHLOG and DFHSHUNT).

See “Defining coupling facility log streams” on page 182 for some sample IXCMIAPU jobs, and see *z/OS MVS Setting Up a Sysplex* for general information about updating LOGR policies.

#### **Define JOURNALMODEL resource definitions**

Define JOURNALMODEL resource definitions in the CICS CSD to enable CICS to map CICS journal names to MVS system logger log stream names. See the *CICS Resource Definition Guide* for information about JOURNALMODEL resource definitions.

#### **Remove JCT definitions**

When migrating a CICS region from a CICS/ESA 4.1 region (or earlier), remove all references to journal control tables (JCTs), and any DD statements for CICS journal data sets, from startup JCL.

See the *CICS Transaction Server for z/OS Migration from CICS TS Version 2.3* for CICS TS Version 1 Releases 1 and 2 for information about obsolete parameters and function relating to the old CICS journal control function and other migration information

#### **Review AKPFREQ system initialization parameter**

When migrating a CICS region from a CICS/ESA 4.1 region (or earlier), review the value specified for AKPFREQ.

This parameter now represents the number of write operations (log records) by CICS log manager to the log stream buffer before an activity keypoint is taken, whereas under the old journal control program it specifies the number of consecutive blocks written to the system log data set.

The parameter has a significant impact on the size of system logger primary (interim) storage, affecting the log tail management that takes place during activity keypoint (AKP) processing. The system logger:

- Deletes records that are no longer of interest to CICS
- Moves records to DFHSHUNT for those tasks that wrote log records within the last ACP.

#### **Update JCL of batch jobs**

When migrating a CICS region from a CICS/ESA 4.1 region (or earlier), update DFHJUP batch job JCL. To process log streams, these jobs require the SUSBSYS keyword on DD statement for the log stream being processed.

### Evaluate results after implementation

After you have implemented the steps necessary to use the MVS system logger for CICS log streams and journals, evaluate the results on a continual basis.

The following are aids that you can use:

- CICS interval statistics. You can collect these at specified intervals and end-of-day to obtain CICS log manager statistics. You can also collect statistics using the DFHOSTAT sample program.
- SMF Type 88 records. These are produced by the MVS system logger, and can be printed using IXGRPT1, which is supplied in SYS1.SAMPLIB. You can also print these records using IXGRPT1J and IXGRPT1L.

The following sections provide more detailed information to help you with the above steps:

- “Coupling facility or DASD-only?” on page 176
- “Coupling facility log streams” on page 177, which contains:
  - “Defining coupling facility structures” on page 178
  - “Planning considerations for the number of log structures” on page 181
  - “Log structure naming conventions” on page 182
  - “Defining coupling facility log streams” on page 182
  - “Sizing considerations for coupling facility log streams” on page 184
  - “Coupling facility requirements in an RLS environment” on page 193
  - “Staging data sets for coupling facility log streams” on page 194.
- “DASD-only log streams” on page 195, which contains:
  - “Defining DASD-only log streams” on page 196
  - “Sizing considerations for DASD-only log streams” on page 197
  - “Converting a DASD-only log stream to use a coupling facility” on page 200.
- “Managing secondary storage” on page 202.

---

## Setting up the environment for CICS log manager

CICS system programmers need to consult with their MVS system programmers to plan for the storage that is required by the log streams needed by the many CICS log managers operating in the sysplex.

Each log stream is a sequence of blocks of data, which the MVS system logger internally partitions over three different types of storage:

1. Primary storage, which holds the most recent records that were written to the log stream. Primary storage can consist of either:
  - a. A structure within a coupling facility. Log data written to the coupling facility is also copied to either a data space or a staging data set.
  - b. A data space in the same MVS image as the system logger. Log data written to the data space is also copied to a staging data set.
2. Secondary storage—when the primary storage for a log stream becomes full, the older records automatically spill into secondary storage, which consists of data sets managed by the storage management subsystem (SMS). Each log stream, identified by its log stream name (LSN), is written to its own log data sets.
3. Tertiary storage—a form of archive storage that is used as specified in your hierarchical storage manager (HSM) policy. Optionally, older records can be migrated to tertiary storage, which can be either DASD data sets or tape volumes.

See the different levels of log stream storage in Figure 9 and Figure 10 on page 176.

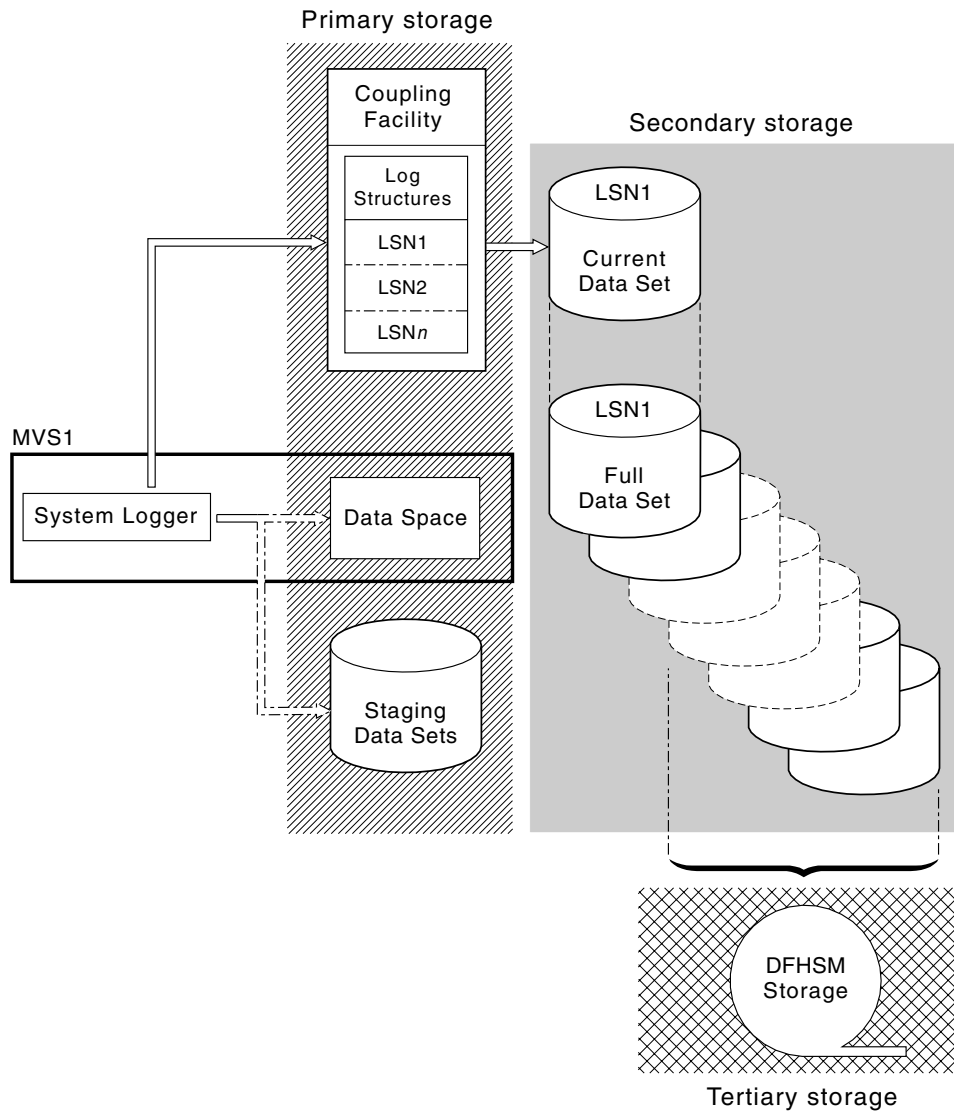


Figure 9. The types of storage used by the MVS system logger. This diagram shows a log stream that uses a coupling facility. Primary storage consists of space in a structure within the coupling facility, and either space in a staging data set or a data space in the same MVS image as the system logger. Secondary storage consists of a series of data sets on disk storage to which the elements of the log structure in the coupling facility are mapped. Tertiary storage is the DFHSM storage holding older levels of the secondary storage data sets.

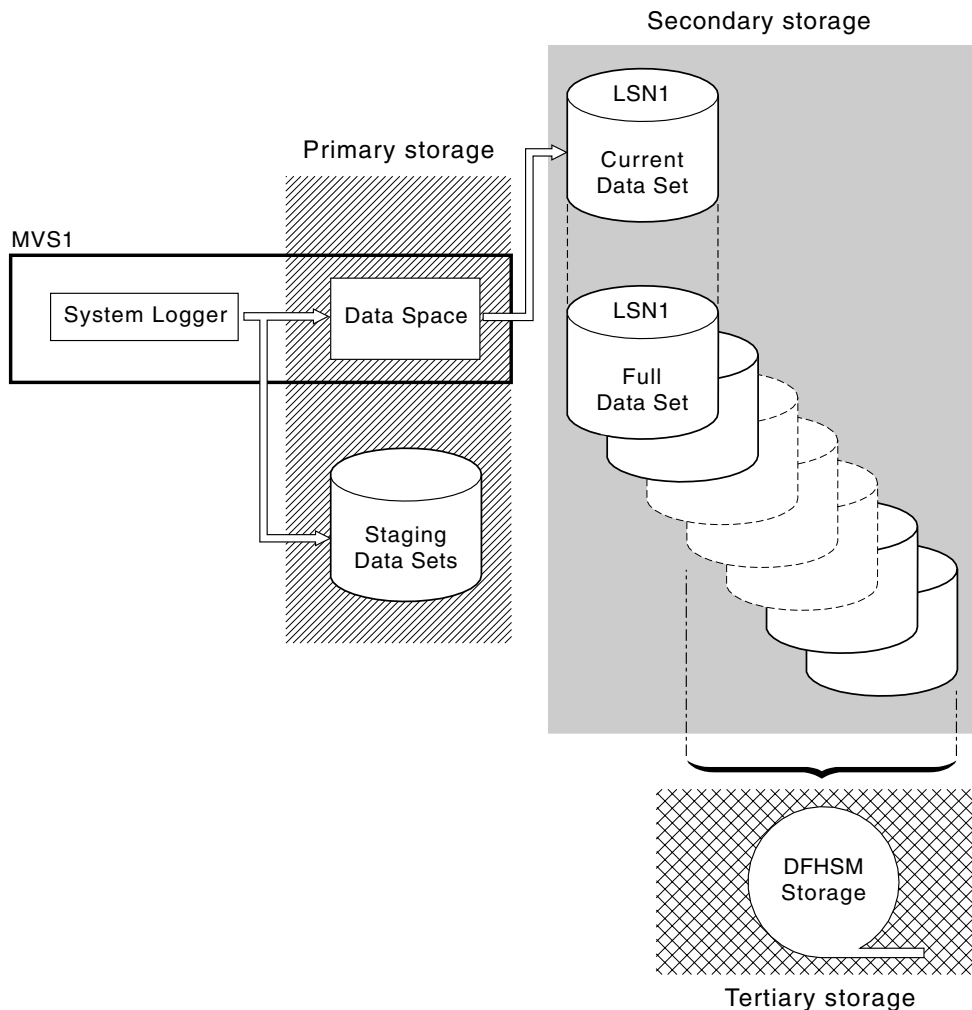


Figure 10. The types of storage used by the MVS system logger. This diagram shows a log stream that uses DASD-only logging. Primary storage consists of a data space in the same MVS image as the system logger, and a single staging data set. Secondary storage consists of a series of data sets on disk storage to which hold successive copies of the single staging data set. Tertiary storage is the DFHSM storage holding older levels of the secondary storage data sets.

## Coupling facility or DASD-only?

The CICS log manager supports the DASD-only option of the MVS system logger. This means that individual CICS log streams can use either coupling facility log structures or DASD-only logging.

Take the following points into account when deciding which log streams should be defined to use the coupling facility and which to use DASD-only:

- A coupling facility log stream must be used if you want to allow simultaneous access from CICS regions running in different MVS images. (Simultaneous access to a DASD-only log stream is limited to CICS regions in the same MVS image.)

For example, assume that you are using RLS and have several CICS application-owning regions (AORs) running on different MVS images. Because the forward recovery log must be accessible from all the AORs, it must be

defined as a coupling facility log stream. A CICS system log, on the other hand, is only ever accessed by a single CICS region, and can therefore always be defined as a DASD-only log stream.

Without a coupling facility, you cannot share general log streams across MVS images.

- Defining all your CICS log streams to use structures within a *single* coupling facility is not recommended—see “Coupling facility log streams.”
- DASD-only log streams are easier to define and administer than coupling facility log streams.
- The CPU cost of a log write to a DASD-only log stream is greater than that of a write to a coupling facility log stream. For more information, see the *CICS Performance Guide*.
- If the amount of available coupling facility space is limited, you may want to define some DASD-only log streams in order to minimize the amount of space allocated to log structures.

**Note:**

1. Define a single-system sysplex (which must use a sysplex couple data set) with PLEXCFG=MONOPLEX. This is required for stand-alone MVS systems that use MVS system logger facilities.
2. Define sysplexes that have two or more MVS images with PLEXCFG=MULTISYSTEM.

---

## Coupling facility log streams

If you use a coupling facility, the ideal environment is provided by two or more non-volatile coupling facilities that are failure-independent from any of the exploiting MVS images, using dedicated processor resources.

Should one coupling facility fail, or require maintenance, in such an environment, the system logger can rebuild its data in another coupling facility and continue. Running CICS systems would experience only minimal impact.

If you are unable to devote two coupling facilities for the purposes of the MVS system logger, the next most robust environment is provided by one dedicated coupling facility for normal logger and lock structure use, plus a coupling facility LPAR. This environment has the same advantages of rebuilding with minimal impact to running CICS systems. Furthermore, MVS detects that the LPAR coupling facility is not in a failure-independent domain, and causes the system logger to write log stream data to staging data sets for extra security.

Running with a single coupling facility is not recommended since its failure would cause the MVS system logger, and any other users of the coupling facility, to suspend normal operation until access to the coupling facility were restored. CICS would, effectively, be unusable in such a situation.

Unless you specify that the system logger is to use staging data sets, the recovery of log stream data depends on the MVS images remaining active so that the system loggers can use copies of log records held in storage to repopulate the coupling facility when it is again available. If you must run with a single coupling facility, you are recommended to specify **DUPLEXMODE(UNCOND)** to force the use of staging data sets.

## Defining coupling facility structures

If you use a coupling facility for your CICS log streams, define the coupling facility structures needed for the log streams in your CFRM policy (in the CFRM data set), and in the LOGR policy (in the LOGR data set).

### Updating the CFRM policy

Coupling facility space is divided into structures using the coupling facility resource management (CFRM) policy defined in the CFRM data set. The CFRM policy allows you to define how MVS is to manage coupling facility resources, and you update this using the IXCMIAPU utility. See Figure 11 on page 179 for a sample job to define coupling facility structures in the CFRM policy data set.

### Updating the LOGR policy

You define structures in the MVS system logger LOGR policy in the system logger couple data sets using the DEFINE STRUCTURE specification of the IXCMIAPU utility. See Figure 12 on page 180 for a sample job to define coupling facility structures in the LOGR policy data set.

## Remember

Before attempting to run any of the IXCMIAPU jobs, ensure that the MVS system logger (IXGLOGR) is running. If IXGLOGR is not running (for example if MVS is running in LOCAL mode), logstream definition jobs fail with rc=0814.



```

//CFRM      JOB (accounting-information),CLASS=A,MSGCLASS=A
//POLICY    EXEC PGM=IXCMIAPU
//STEPLIB  DD DSN=SYS1.MIGLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//*****
//*
/* System logger structure definitions should be merged with      *
/* definitions for other structures required by the sysplex.      *
/*
/* Space values are for illustration only -- substitute values    *
/* appropriate to your number of logs and expected activity.      *
/*
/* NOTE: The values in this job are not matched with the other    *
/*       sample jobs.                                             *
//*****
//SYSIN     DD *
DATA TYPE(CFRM) REPORT(YES)
DELETE POLICY NAME(POL1)
DEFINE POLICY NAME(POL1)
  PREFLIST(cfname) REBUILDPERCENT(1)
/* Define coupling facilities */
CF NAME(cfname)
  TYPE(009674)
  MFG(IBM)
  PLANT(00)
  SEQUENCE(000000040032)
  PARTITION(3)
  CPCID(00)
  DUMPSPACE(8192)
/* Define other structures required by sysplex here . . .      */
...
/* Define logger structures for CICS log streams.                */
/* - If a choice of facilities is available, use non-volatile    */
/*   facilities if possible                                     */
/* - Specify low REBUILDPERCENT so that structures are rebuilt  */
/*   in the event of connectivity failure                       */
/* - INITSIZE gives initial coupling facility size (based on    */
/*   sizing calcs)                                           */
/* - SIZE should be larger to allow for rebuild to a larger    */
/*   size if INITSIZE proves to be too small                 */
/* - SIZE and INITSIZE values are for illustration only -     */
/*   substitute values appropriate for your intended usage.    */

STRUCTURE NAME(LOG_DFHLOG_001)      /* CICS system logs */
INITSIZE(10000) SIZE(16000)
PREFLIST(cfname) REBUILDPERCENT(1)

STRUCTURE NAME(LOG_DFHSHUNT_001)    /* CICS secondary logs */
INITSIZE(10000) SIZE(16000)
PREFLIST(cfname) REBUILDPERCENT(1)

STRUCTURE NAME(LOG_USERJNL_001)     /* CICS user journals */
INITSIZE(10000) SIZE(16000)
PREFLIST(cfname) REBUILDPERCENT(1)

STRUCTURE NAME(LOG_GENERAL_001)     /* Forward recovery logs */
INITSIZE(10000) SIZE(16000)
PREFLIST(cfname) REBUILDPERCENT(1)
/*
//

```

Figure 11. Sample policy job to define logger structures to CFRM 1/2

Multiple log streams can write data to a single coupling facility structure. This does not mean that the log data is merged; the log data stays segregated according to

log stream. You can specify the number of log streams that use the resources of a single coupling facility structure using the LOGSNUM parameter on the IXCMIAPU service to define a structure.

Each log stream is allocated a proportion of the structure space based on the number of currently connected log streams (up to the limit specified in LOGSNUM).

For example, a structure may be defined to contain a maximum of, say, 30 log streams. If only 10 log streams are connected, each log stream can use one tenth of the space in the structure. As other log streams are connected and disconnected, the MVS system logger adjusts the proportion of space to be used by each log stream.

It is important to plan carefully before specifying a value for LOGSNUM, because this parameter determines how much storage space in the structure is available to each log stream. A number in the range 10 to 20 is optimum in many environments.

The JCL in Figure 12 defines log stream coupling facility structures to the MVS system logger. It is meant for guidance only and you should substitute values appropriate to your requirements.

```
//DEFSTRUC JOB ...
//POLICY EXEC PGM=IXCMIAPU
//STEPLIB DD DSN=SYS1.MIGLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//*****
//*
/* Define log stream coupling facility structures to the MVS logger *
/*
/* AVGBUFSIZE and LOGSNUM values are just for illustration, *
/* substitute values appropriate to your intended usage *
/*
//*****
//SYSIN DD *
DATA TYPE(LOGR) REPORT(YES)

/* System logs */
DEFINE STRUCTURE NAME(LOG_DFHLOG_001) LOGSNUM(10)
MAXBUFSIZE(64000) AVGBUFSIZE(500)

/* Secondary system logs */
DEFINE STRUCTURE NAME(LOG_DFHSUNT_001) LOGSNUM(10)
MAXBUFSIZE(64000) AVGBUFSIZE(4096)

/* User journals with unforced writes */
DEFINE STRUCTURE NAME(LOG_USERJRNL_001) LOGSNUM(10)
MAXBUFSIZE(64000) AVGBUFSIZE(64000)

/* Fwd recovery logs and user jnl's that are forced */
DEFINE STRUCTURE NAME(LOG_GENERAL_001) LOGSNUM(10)
MAXBUFSIZE(64000) AVGBUFSIZE(2048)

/*
//
```

Figure 12. Sample JCL to define coupling facility structures to MVS system logger

See the *z/OS MVS Programming: Assembler Services Guide* for information on planning coupling facility configurations.

## Planning considerations for the number of log structures

Bear in mind the following points when planning the definition of your coupling facility structures:

- The CFRM policy allows a maximum of 255 structures for all purposes.
- Allow a maximum of 20 log streams per structure.
- Smaller structures are more quickly allocated, rebuilt, and recovered than larger ones.
- It is good practice to keep the log streams for test CICS systems (and other systems not in regular use) in structures separate from the structures holding the log streams of production CICS systems. This avoids the structure space available to production CICS systems being affected by structure usage of the test CICS systems.
- It is good practice to keep the log streams for terminal-owning regions (TORs) in structures separate to those accommodating log streams for application-owning regions (AORs). In addition, keep log streams for file-owning regions in structures separate to those accommodating log streams for TORs and AORs.
- Share structures between MVS images. If an MVS image or logger address space fails, and a surviving MVS image is using the same log stream structures (although not necessarily the same log streams), the surviving image is notified of the failure and can initiate immediate log stream recovery for the failing MVS. Recovery would, otherwise, be delayed until the next time that a system attempts to connect to a log stream in the affected structures, or until the logger address space of the failing system is restarted.

For example, in a 4-way sysplex comprising MVSA, MVSB, MVSC, and MVSD, you might have the CICS regions that normally run on MVSA and MVSB use structure LOG\_DFHLOG\_001, and the regions that run on MVSC and MVSD use structure LOG\_DFHLOG\_002. Thus each MVS image has a partner to recover its log streams in the event of an MVS failure. If a structure fails, the two MVS images using the other structure can take over the workload. Also, if you have more than one coupling facility, allocate the system log structures to different coupling facilities. See Figure 13 for an illustration of this example.

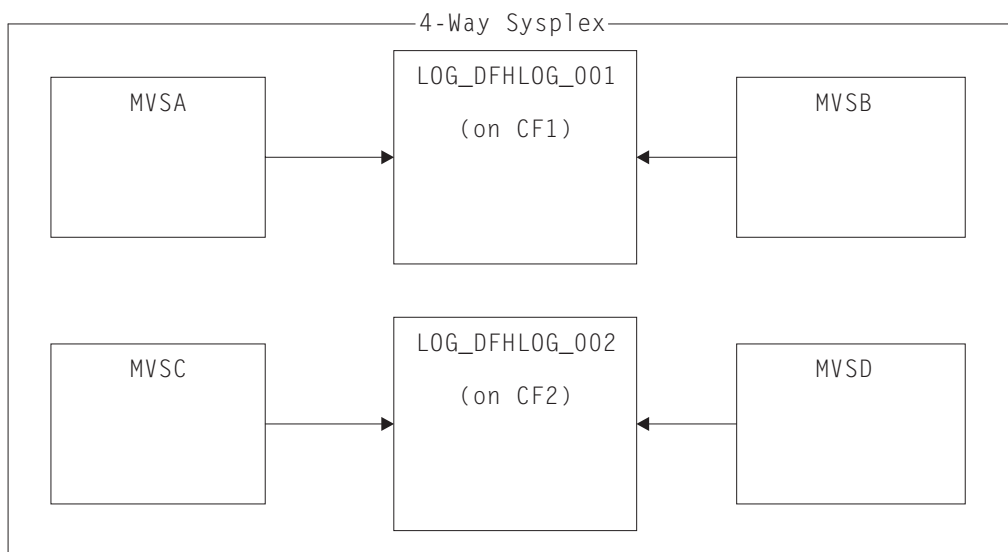


Figure 13. Sharing system logger structures between MVS images

- Use the appropriate buffer size. The average buffer size (AVGBUFSIZE) defined for a structure should be reasonably close to the actual buffer size of the log streams using the structure. If it is not, there is a risk that usable space will be exhausted long before the structure is actually full.

**Important:**

1. OS/390 (since Release 3) dynamically tunes the element/entry ratio, so the value you specify for AVGBUFSIZE is less important than it was on earlier releases of MVS.
  2. AVGBUFSIZE, like other structure definition attributes such as MAXBUFSIZE and LOGSNUM, cannot be updated unless you first delete the log streams in the structure definition.
- Set MAXBUFSIZE to slightly less than 64K - say, 64000. This allows CICS to write the maximum size user record and allows coupling facility storage to be allocated in 256-byte units. If you allow MAXBUFSIZE to default, coupling facility storage is allocated in 512-byte units. This can be wasteful of storage.  
There is no significant advantage in setting MAXBUFSIZE lower than 64000 as far as the utilization of storage is concerned.
  - Set a low value for the REBUILDPERCENT parameter in the CFRM policy for log structures used for CICS system logs.

## Log structure naming conventions

It is sensible to adopt a naming convention for your coupling facility structures that help to identify the purpose of the structure. A format such as LOG\_*purpose\_nnn* is recommended, where:

- *purpose* identifies the type of use of the structure.
- *nnn* is a sequence number to allow for more than one structure for each purpose.

Some examples are:

**LOG\_DFHLOG\_001**

For the CICS primary system log. The structure should be large to avoid the need to write data to DASD. The average buffer size would be small. See the sizing calculations in “Structure size for system log usage” on page 187.

**LOG\_DFHSUNT\_001**

For the CICS secondary system log. The structure should be small but requires a large buffer size. A structure of 150K per log stream may well be sufficient.

**LOG\_USERJRNL\_001**

For user journals where block writes are not forced. The average and maximum buffer sizes of these structures should be the same.

**LOG\_GENERAL\_001**

For forward recovery logs and user journals where block writes are forced periodically.

See also the section “Develop a naming convention for system logger resources” in the *z/OS MVS Setting Up a Sysplex* manual.

## Defining coupling facility log streams

Use the MVS IXCMIAPU utility to define coupling facility log streams to the LOGR couple data set. The basic syntax to define a coupling facility log stream is as follows:

```

DEFINE LOGSTREAM NAME(log_stream_name)
                STRUCTNAME(structure_name)
                LOWOFFLOAD(low_offload) HIGHOFFLOAD(high_offload)
                STG_DUPLEX(YES|NO) DUPLEXMODE(COND|UNCOND)

```

For detailed information about the full range of log stream attributes, see the *z/OS MVS Setting Up a Sysplex* manual. Figure 14 shows example definitions for a pair of coupling facility log streams associated with a CICS system log.

```

//DEFLOGS JOB ...
//LOGDEFN EXEC PGM=IXCMIAPU
//STEPLIB DD DSN=SYS1.MIGLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//*****
//*
//* Define coupling facility log streams for CICS system log.
//*
//* The LOWOFFLOAD value is for illustration only --
//* substitute a value appropriate for your environment.
//*
//*****
//SYSIN DD *
DATA TYPE(LOGR) REPORT(NO)
DEFINE LOGSTREAM NAME(region_userid.applid.DFHLOG)
                STRUCTNAME(LOG_DFHLOG_001)
                LOWOFFLOAD(40) HIGHOFFLOAD(80)
                STG_DUPLEX(YES) DUPLEXMODE(COND)
DEFINE LOGSTREAM NAME(region_userid.applid.DFHSUNT)
                STRUCTNAME(LOG_DFHSUNT_001)
                LOWOFFLOAD(40) HIGHOFFLOAD(80)
                STG_DUPLEX(YES) DUPLEXMODE(COND)

```

Figure 14. Example definitions of coupling facility log streams. The definitions are for the CICS primary and secondary system log streams. The value **region\_userid** is the RACF userid under which the CICS address space is running; **applid** is the CICS region's VTAM APPL name (taken from the APPLID system initialization parameter).

## Using model log streams

To avoid having to define explicitly each log stream used by each of your CICS regions, you can use model log stream definitions. Using models, log streams are defined to MVS dynamically, on their first usage. Figure 15 on page 184 shows an example of coupling facility model definitions for CICS primary and secondary system log streams.

```

//DEFLOGS JOB ...
//LOGDEFN EXEC PGM=IXCMIAPU
//STEPLIB DD DSN=SYS1.MIGLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//*****
//*
//* Define coupling facility model log streams for CICS system log.*
//*
//* The LOWOFFLOAD value is for illustration only --
//* substitute a value appropriate for your environment.
//*
//*****
//SYSIN DD *
DATA TYPE(LOGR) REPORT(NO)
DEFINE LOGSTREAM NAME(sysname.DFHLOG.MODEL)
MODEL(YES)
STRUCTNAME(LOG_DFHLOG_001)
LOWOFFLOAD(40) HIGHOFFLOAD(80)
STG_DUPLEX(YES) DUPLEXMODE(COND)
DEFINE LOGSTREAM NAME(sysname.DFHSUNT.MODEL)
MODEL(YES)
STRUCTNAME(LOG_DFHSUNT_001)
LOWOFFLOAD(40) HIGHOFFLOAD(80)
STG_DUPLEX(YES) DUPLEXMODE(COND)

```

Figure 15. Example model definitions for coupling facility system log streams. The value **sysname** is the sysid of the MVS image in which the CICS region or regions are running.

For detailed information about using model log streams, see the *CICS Recovery and Restart Guide*. For information about the mapping of CICS journal definitions to log stream names, see the *CICS System Definition Guide*.

When using model log streams, you need to bear the following in mind:

- For coupling facility log streams, a model log stream definition determines the coupling facility structure in which the new log streams are created. On an MVS image that runs both CICS production and CICS test regions, take care that the system logs for the production regions are kept separate from the system logs for the test regions.
- There are recovery considerations when using model log streams to define CICS system logs—see the *CICS Recovery and Restart Guide* for more information.

## Sizing considerations for coupling facility log streams

This section discusses how to size the following types of coupling facility log stream:

- The CICS primary and secondary system log streams
- Forward recovery logs
- User journals and autojournals.

### Sizing DFHLOG

For the CICS primary system log stream (DFHLOG), it is important to:

- **Minimize the amount of data that is offloaded to secondary storage:**

The MVS system logger begins the offload process when the high offload threshold (HIGHOFFLOAD) of the log stream is reached. The offload process consists of two steps:

1. The MVS logger physically deletes the data in the log stream that has been marked for deletion by the CICS log-tail deletion process.

2. The MVS logger calculates how much data needs to be offloaded to secondary storage, based on the difference between HIGHOFFLOAD and LOWOFFLOAD, less the amount of data that has been deleted since the last offload event.

To minimize the amount of data offloaded from the CICS primary system log:

- Define a suitably-sized coupling facility structure. For advice, see “Setting attribute values for structure, log stream and system definitions.”
- Ensure that the log-tail deletion process is working effectively. For detailed information about the log tail deletion process, see the *CICS Recovery and Restart Guide*.

- **Avoid “structure-full” events:**

A structure-full event occurs when a log stream's structure space becomes full before the offloading of data has completed.

For advice on monitoring and avoiding structure-full events, see the *CICS Performance Guide*.

## Sizing DFHSHUNT

It is important to size the secondary system log stream (DFHSHUNT) to avoid structure-full events. However, it is normal for some data to be offloaded from DFHSHUNT to secondary storage.

## Sizing general logs

It is important to size forward recovery logs, user journals, and autojournals to avoid structure-full events. However, because CICS does not delete data from these log streams, it is normal for data to be offloaded to secondary storage.

## Setting attribute values for structure, log stream and system definitions

You can use the System z<sup>®</sup> Coupling Facility Structure Sizer tool (CFSizer), or the formulas provided, to calculate storage requirements for coupling facility log streams.

CFSizer is a web-based application that communicates with a coupling facility at a current CFLEVEL to calculate storage requirements. See CFSizer.

You can use the CFSizer tool to calculate the following values:

- INITSIZE
- SIZE

For more information, see Structure size for system log usage.

For information about the CPC support for different CFLEVELs and the function in each CFLEVEL, see CF levels. For more information about the storage increments for different CFLEVELs, see the information about coupling facility control code support in *System z10<sup>®</sup> Processor Resource/Systems Manager Planning Guide*.

You can use the formulas provided in Structure size for system log usage to calculate the following values:

- INITSIZE
- AVGBUFSIZE
- SIZE
- LOWOFFLOAD

You can use the formula provided in Staging data set size calculation to calculate the following value:

- STG\_SIZE

You must base your calculations on the journaling requirements of your applications.

Table 11 summarizes how you decide on the values for various attributes on the structure definition, log stream definition, and system definition.

*Table 11. How to decide on the values of attributes*

Facility	Attribute	Method or Value
Structure	INITSIZE	Use CFSizer or the formula in INITSIZE calculation.
	SIZE	Use CFSizer or the formula in SIZE calculation.
	AVGBUFSIZE	Use the formula in AVGBUFSIZE calculation. Underestimate rather than overestimate.
	MAXBUFSIZE	64 000
Primary system log stream (DFHLOG)	HIGHOFFLOAD	80
	LOWOFFLOAD	Use the formula in LOWOFFLOAD calculation.
Secondary system log stream (DFHSHUNT)	HIGHOFFLOAD	80
	LOWOFFLOAD	0
General log stream	HIGHOFFLOAD	80
	LOWOFFLOAD	40 – 60
Log stream	STG_SIZE	Use the formula in Staging data set size calculation. Overestimate rather than underestimate.
CICS system	AKPFREQ	4 000

Startup might take longer than you experienced when using earlier releases. This is partly caused by the allocation and formatting of the staging data sets. Increased startup time depends on factors that include the following:

- Size of staging data set (STG\_SIZE)
- DASD speed
- DASD contention

You can reduce startup time by avoiding large staging data sets.

### Sizing log streams

If you want to size your coupling facility structures and log streams, the following sections help you to calculate your space requirements.

The formulae provided help you to calculate values for:

- INITSIZE
- AVGBUFSIZE
- SIZE
- LOWOFFLOAD
- STG\_SIZE



You must base your calculations on the journaling requirements of your applications. This provides the starting point for the following formulae.

### **Structure size for system log usage:**

You can use the System z Coupling Facility Structure Sizer tool (CFSizer), or the formulas provided, to calculate storage requirements for the primary system log, DFHLOG, and the secondary system log, DFHSHUNT.

The CFSizer calculations assume that the coupling facility is at the current CFLEVEL. The formulas provided assume the coupling facility is at CFLEVEL 11.

You use the results of these calculations to set the coupling facility resource manager (CFRM) **INITSIZE** and **SIZE** parameters in the CFRM policy.

The primary and secondary log streams must be placed in different structures because of the large disparity in data volumes written to the primary and secondary system logs.

Generally, the volume of data that CICS keeps in the primary system log at any one time is slightly greater than the amount written during one activity keypoint interval. This volume is determined by the activity keypoint frequency, which is measured in the number of write requests to the CICS system logstream output buffer, and defined on the **AKPFREQ** system initialization parameter. When you plan coupling facility structure sizes, review the value specified for the **AKPFREQ** parameter.

### **Using CFSizer to determine structure size**

CFSizer is a web-based application that communicates with a coupling facility at a current CFLEVEL to calculate storage requirements. For primary and secondary system logs, it calculates suitable values for the **INITSIZE** and **SIZE** parameters. See <http://www.ibm.com/systems/support/z/cfsizer/>.

To use CFSizer to calculate storage requirements for system logs, enter the following information:

#### **Maximum buffer size**

The size, in bytes, of the largest log block that can be written to a logstream. The value must be between 1 and 65532. This value corresponds to the **MAXBUFSIZE** value in the definition of the logstream coupling facility structure. See “Defining coupling facility structures” on page 178.

#### **Average buffer size**

The average size in bytes of log blocks written to all logstreams. This input is used to determine the initial entry to element ratio of the list structure. This value corresponds to the **AVGBUFSIZE** value in the definition of the logstream coupling facility structure. See “Defining coupling facility structures” on page 178.

Also, this value is available in the system logger reports produced by IXGRPT1 and IXGRPT1J. See “Analyzing SMF Type 88 records” on page 201.

#### **Highoffload**

The high offload threshold of the logstream. When this percentage of the coupling facility structure is full, the system logger begins to offload data from primary storage to offload data sets. For CICS logstreams, this value

is usually between 80 and 85. See Checking logstream status and LOWOFFLOAD and HIGHOFFLOAD parameters on log stream definition in *CICS Performance Guide*.

For more information about the offload threshold parameter, see *z/OS MVS Setting Up a Sysplex*.

#### **Number of write requests per interval**

The number of log blocks written to the logstream (the total from all systems that use the logstream) during the data collection interval. This value is available in the CICS statistics reports. See Logstream statistics and Logstreams reports in *CICS Performance Guide*.

Also, this value is available in the #WRITES INVOKED data in the system logger reports produced by IXGRPT1 and IXGRPT1J. See “Analyzing SMF Type 88 records” on page 201.

#### **Length of interval in seconds**

The time interval in seconds over which the number of writes was collected. This value is available in the CICS statistics reports. See Statistics domain: Summary global statistics in *CICS Performance Guide*

#### **Longest running task in seconds**

The duration in seconds of the longest running transaction during the data collection interval. This is not the average response time.

#### **Log tail deletes**

The number of times CICS performed log tail deletion during the data collection interval. This value is available in the CICS statistics reports. See delete requests, stream deletes or logstream deletes in Logstream statistics and Logstreams reports in *CICS Performance Guide*. You can also determine this value by using joblog of the region and counting the number of DFHLG0743 messages that were issued during the interval.

#### **Calculating INITSIZE for DFHLOG**

To calculate the INITSIZE value to use in the CFRM policy for the primary system log, use the following formula:

$$\text{INITSIZE} = 310 + (\text{LOGSNUM} * \text{A} * \text{B} / 1024)$$

where:

- A = 2000 + (number\_of\_entries + 5)
- B = (AVGBUFSIZE \* 1.1289) + 195

To calculate a value for number\_of\_entries, use the following formula:

$$\text{number\_of\_entries} = ((\text{akpintvl} + \text{trandur}) * \text{writespersec}) / 0.9$$

where:

- akpintvl is the interval between activity key points, which varies with workload. Calculate it as follows:

$$\text{akpintvl} = \text{AKPFREQ} / ((\text{N1} * \text{R1}) + (\text{N2} * \text{R2}) + (\text{Nn} * \text{Rn}))$$

where:

- N1, N2 . . . . Nn is the transaction rate for each transaction (transactions per second).
- R1, R2 . . . . Rn is the number of log records written by each transaction.

- `trandur` is the execution time (between sync points) of the longest-running transaction that runs as part of the normal workload.

If this duration is longer than the `akpintvl` value, you have the following choices:

- Increase the value of the `AKPFREQ` parameter, thus increasing the value of `akpintvl`, provided that the value does not result in an unacceptably large coupling facility structure size.
- Change the application logic to cause more frequent sync points.
- Calculate a structure size based on a shorter transaction duration, and accept that DASD offloading occurs when the long-running transaction is used.
- `writespersec` = lesser of 25 or  $((N1 * R1) + \dots (Nn * Rn))$ , where:
  - `N1, N2 . . . . Nn` are the transaction frequencies (transactions per second) of the most frequently run transactions.
  - `R1, R2 . . . . Rn` is the number of log records written by each transaction.

To calculate the value of `AVGBUFSIZE` for the primary system log, `DFHLOG`, you can use the weighted average of the data logged by the most frequently run transactions in the system:

$$\text{AVGBUFSIZE} = (\text{bytespersec} / \text{writespersec}) + 48$$

where:

- `bytespersec` =  $(N1 * D1) + (N2 * D2) + \dots (Nn * Dn)$ 
  - `N1, N2 . . . . Nn` are the transaction frequencies (transactions per second) of the most frequently run transactions.
  - `D1, D2 . . . . Dn` are the bytes of data logged by each transaction.

You can calculate the amount of data (`Dn`) written to the system log for each transaction:

$$\begin{aligned} Dn = & Ns * \text{syncreclen} + \\ & Nfc * (\text{fcrechdr} + \text{fcreclen}) + \\ & Nts * (\text{tsrechdr} + \text{tsreclen}) + \\ & Ntd * (\text{tdrechdr} + \text{tdreclen}) + \\ & Nur * (\text{urrechdr} + \text{urreclen}) \end{aligned}$$

where:

- `Ns` is the number of sync points per transaction; usually 1.
- `syncreclen` is the sync point record length.
- `Nfc, fcrechdr, fcreclen` are the number of recoverable updates made, the length of the record headers, and the length of the records for file control, respectively.  
Count only READ UPDATE and WRITE ADD records. `fcrechdr` is 144 (136 bytes of record header plus 8 bytes of file name).
- `Nts, tsrechdr, tsreclen` are for recoverable temporary storage updates.  
Count only TS PUT and TS UPDATE records. For TS PUT records, `tsrechdr` is 108, and `tsreclen` is 88. For TS UPDATE records, `tsrechdr` is 108, and `tsreclen` is 52.
- `Ntd, tdrechdr, tdreclen` are for recoverable transient data updates.  
`tdrechdr` is 108, and `tdreclen` is 380.
- `Nur, urrechdr, urreclen` are for user records written to `DFHLOG`.  
`urrechdr` is 125.
- See Writes per second calculation for details of how to calculate `writespersec`.

If the result of the calculation shows a value for AVGBUFSIZE that is greater than the value defined for MAXBUFSIZE, the value defined for MAXBUFSIZE is taken as the value for AVGBUFSIZE, and writespersec is calculated as follows:

$$\text{writespersec} = \text{bytespersec} / (\text{MAXBUFSIZE} - 48)$$

Round the final result of the INITSIZE formula up to the next multiple of 256.

### Calculating SIZE for DFHLOG

To calculate the SIZE value to use in the CFRM policy for the primary system log, use the following formula:

$$\text{SIZE} = 480 + (\text{LOGSNUM} * \text{A} * \text{B} / 1024)$$

where:

- $\text{A} = 2500 + (\text{number\_of\_entries} + 5)$
- $\text{B} = (\text{AVGBUFSIZE} * 1.6821) + 289$

To calculate a value for number\_of\_entries, see number of entries calculation.

Round the final result of the SIZE formula up to the next multiple of 256. The formula for SIZE gives a result that is approximately fifty percent greater than the INITSIZE value.

### Calculating LOWOFFLOAD for DFHLOG

You can calculate a suitable value for LOWOFFLOAD for DFHLOG using one of the following formulas:

- $\text{LOWOFFLOAD} = (\text{trandur} * 90) / (\text{akpintvl} + \text{trandur}) + 10$  (where RETPD=0 specified)
- $\text{LOWOFFLOAD} = (\text{trandur} * 90) / (\text{akpintvl} + \text{trandur})$  (where RETPD=dddd specified)

where:

- akpintvl is the interval between activity key points. See activity key point interval calculation.
- trandur is the execution time, between sync points, of the longest-running transaction that runs as part of the normal workload.

If this duration is longer than akpintvl value, you have the following choices:

- Increase the value of the **AKPFREQ** parameter, thus increasing the value of akpintvl, provided that this value does not result in an unacceptably large coupling facility structure size.
- Change the application logic to cause more frequent sync points.
- Calculate a structure size based on a shorter transaction duration, and accept that DASD offloading occurs when the long-running transaction is used.

For a description of the low offload threshold for a logstream, see LOWOFFLOAD and HIGHOFFLOAD parameters on log stream definition in *CICS Performance Guide*.

### Calculating INITSIZE and SIZE for DFHSHUNT

Generally, the secondary system logstream is only a fraction of the size of the primary logstream. To calculate coupling facility space for DFHSHUNT, use the following formulas:

- $INITSIZE = (150 * LOGSNUM) + 310$
- $SIZE = (230 * LOGSNUM) + 480$

where LOGSNUM is number of logstreams that can write data to the secondary logstream.

**Structure size for forward recovery log usage:** You can merge the forward recovery logs written by many CICS regions onto the same log stream. You can also use the same log stream for forward recovery data for multiple data sets.

See “Structure size for system log usage” on page 187 and “Structure size for system log usage” on page 187 for the formulae to calculate values for the **INITSIZE** and **SIZE** attributes.

Calculate a value for number of entries as follows:

$$\text{no. entries} = \text{writespersec} * 12.5$$

where:

$$\text{writespersec} = \text{lesser of } 25 \text{ or } (N1 + \dots + Nn)$$

where N1 . . . . Nn is the number of transactions per second writing to each data set.

You can calculate **AVGBUFSIZE** as follows:

$$\text{AVGBUFSIZE} = (\text{bytespersec} / \text{writespersec}) + 36$$

where:

- $\text{bytespersec} = (N1 * Wr1 * (D1 + \text{rechdr}) + \dots (Nn * Wrn * (Dn + \text{rechdr})))$
- $\text{writespersec} = \text{lesser of } 25 \text{ or } (N1 + \dots + Nn)$ , where:
  - N1 . . . . Nn is the number of transactions per second writing to each data set.
  - Wr1 . . . Wrn is the number of write requests per transaction.
  - D1 . . . . Dn is the average record length for each data set.
- rechdr is the record header length of each record.

If the records are WRITE ADD, WRITE ADD COMPLETE, or WRITE ADD DELETE records, rechdr is **84** and is followed by the record key, and the record data (including its key).

If the result of the calculation shows a value for AVGBUFSIZE that is greater than the value defined for MAXBUFSIZE, then the value defined for MAXBUFSIZE is taken as the value for AVGBUFSIZE, and writespersec is calculated as follows:

$$\text{writespersec} = \text{bytespersec} / (\text{MAXBUFSIZE} - 36)$$

**Structure size for user journal and autojournal usage:**

You can use the System z Coupling Facility Structure Sizer tool (CFSizer), or the formulas provided, to calculate storage requirements for the user journal and autojournal.

The CFSizer calculations assume that the coupling facility is at the current CFLEVEL. The formulas provided assume the coupling facility is at CFLEVEL 11.

## Using CFSizer to determine structure size

CFSizer is a web-based application that communicates with a coupling facility at a current CFLEVEL to calculate storage requirements. For the user journal and autojournal, it calculates suitable values for the **INITSIZE** and **SIZE** parameters. See CFSizer.

To use CFSizer to calculate storage requirements for the user journal and autojournal, enter the following information:

### **Writes per second**

The number of log blocks written to the log stream per second from a single system. This value is available in the CICS statistics reports. See Logstreams reports in *CICS Performance Guide*.

### **Maximum buffer size**

The size, in bytes, of the largest log block that can be written to a logstream. The value must be between 1 and 65532. This value corresponds to the MAXBUFSIZE value in the definition of the logstream coupling facility structure. See “Defining coupling facility structures” on page 178.

### **Average buffer size**

The average size in bytes of log blocks written to all logstreams. This input is used to determine the initial entry to element ratio of the structure. This value corresponds to the AVGBUFSIZE value in the definition of the logstream coupling facility structure. See “Defining coupling facility structures” on page 178.

### **Number of logstreams**

The number of logstreams allocated to a structure. This must be a value from 0 to 512. In a CICS environment, the number of logstreams should be between 10 and 20.

For the user journal and autojournal, CFSizer uses the following default values:

### **Highoffload**

80

### **Lowoffload**

0

## Using formulas to calculate structure size

Calculate values for the **INITSIZE** and **SIZE** attributes. See “Calculating **INITSIZE** for DFHLOG” on page 188 and “Calculating **SIZE** for DFHLOG” on page 190.

To calculate a value for `number_of_entries`, use the following formula:

$$\text{number\_of\_entries} = \text{writespersec} * 12.5$$

See the explanation of `writespersec` later in this topic.

For journals with log blocks that are not forced to the log stream, the average block size tends to be slightly less than the **MAXBUFSIZE** value defined for the coupling facility structure.

For journals with log blocks that are forced to the log, using the **EXEC CICS WAIT JOURNALNAME** or **EXEC CICS WAIT JOURNALNUM** commands, or using the **WAIT** option of the **EXEC CICS WRITE JOURNALNAME** or **EXEC CICS WRITE JOURNALNUM** commands, you

can calculate AVGBUFSIZE from the weighted average of the data logged for each journal logging to the same log stream for a given CICS system.

$$\text{AVGBUFSIZE} = (\text{bytespersec} / (\text{writespersec}) + 36$$

where:

- $\text{bytespersec} = (N1 * Wr1 * (D1 + \text{rechdr}) + \dots (Nn * Wrn * (Dn + \text{rechdr})))$
- $\text{writespersec} = \text{lesser of } 25 \text{ or } ((N1 * Wa1) + \dots + (Nn * Wan))$  where:
  - $N1, \dots, Nn$  is the number of transactions per second writing to the journal.
  - $Wr1 \dots Wrn$  is the number of write requests per transaction.
  - $Wa1 \dots Wan$  is the number of wait requests per transaction.
  - $D1 \dots Dn$  is the average record length of each journal record.
  - $\text{rechdr}$  is the record header length of each record.

Autojournal records are issued from file control. They might be DATA SET NAME records, which consist of a 204-byte record header, and no further data. Alternatively, they might be READ ONLY, READ UPDATE, WRITE UPDATE, WRITE ADD, or WRITE ADD COMPLETE records. In this case,  $\text{rechdr}$  is 84 bytes and is followed by the file control record itself.

User journal records consist of a 68-byte record header, followed by the user prefix and the user data.

If the result of the calculation shows a value for AVGBUFSIZE that is greater than the value defined for MAXBUFSIZE, the value defined for MAXBUFSIZE is taken as the value for AVGBUFSIZE, and writespersec is calculated as follows:

$$\text{writespersec} = \text{bytespersec} / (\text{MAXBUFSIZE} - 36)$$

## Coupling facility requirements in an RLS environment

When you move to an RLS environment from an environment in which multiple AORs have been accessing data sets in a FOR, the logging activity of the FOR is distributed across the AORs. Therefore, the coupling facility structure size required by each AOR increases.

You can use the formulae for INITSIZE and SIZE, given in “Calculating INITSIZE for DFHLOG” on page 188 and “Calculating SIZE for DFHLOG” on page 190. However, you need to use different formulas to calculate the following values:

- $\text{avgbufsize}$
- number of entries
- $\text{akpintvl}$ .

Use the reports produced the log stream statistics from CICS Transaction Server for z/OS, to calculate the number of log write operations and the amount of data written in a reporting interval for the AORs and the FOR.

### Calculating increased AOR coupling facility storage requirements

Use the following formulae to calculate:

- $\text{avgbufsize}$
- number of entries
- $\text{akpintvl}$ .

for the AORs in the new RLS environment.

Calculate the AOR AVGBUFSIZE value required by the INITSIZE and SIZE formulae as follows:

$$\text{AOR AVGBUFSIZE} = ( \text{AOR\_bytes} + ( \text{FOR\_bytes} / \text{no. of AORs} ) ) / ( \text{intvlen} * 25 )$$

where:

- AOR\_bytes is the number of bytes written to the system log by an AOR in the sampling interval.
- FOR\_bytes is the number of bytes written to the system log by an FOR in the sampling interval.
- number\_of\_AORs is the number of cloned AORs using the FOR.
- intvlen is the length (in seconds) of the sampling interval (statistics).

Calculate the AOR number of entries value required by the INITSIZE and SIZE formulas as follows:

$$\text{AOR number\_of\_entries} = ( (\text{AOR\_akpintvl} + \text{trandur}) * 25 ) / 0.9$$

where:

- 

$$\text{AOR\_akpintvl} = ( \text{AKPFREQ} * \text{intvlen} ) / ( \text{AOR\_recs} + ( \text{FOR\_recs} / \text{no. of AORs} ) )$$

where:

- intvlen is the length (in seconds) of the sampling interval (statistics).
- AOR\_rec is the number of records written to the system log by an AOR in the sampling interval.
- FOR\_rec is the number of records written to the system log by an FOR in the sampling interval.
- number\_of\_AORs is the number of cloned AORs using the FOR.
- trandur is the execution time (between syncpoints) of the longest-running transaction that runs as part of the normal workload.

If this is longer than AOR\_akpintvl, use AOR\_akpintvl as the duration or consider increasing AKPFREQ.

Once you have calculated the values for AOR AVGBUFSIZE and AOR number\_of\_entries, use the formulae for INITSIZE and SIZE, as described in “Calculating INITSIZE for DFHLOG” on page 188 and “Calculating SIZE for DFHLOG” on page 190.

## Staging data sets for coupling facility log streams

MVS normally keeps a second copy of the data written to the coupling facility in a data space, for use when rebuilding a coupling facility log in the event of an error. This is satisfactory as long as the coupling facility is failure-independent (in a separate CPC and non-volatile) from MVS.

Where the coupling facility is in the same CPC, or uses volatile storage, the MVS system logger supports staging data sets for copies of log stream data that would otherwise be vulnerable to failures that impact both the coupling facility and the MVS images.

The following **recommendations** are for guidance when defining log streams:

- Define STG\_DUPLEX(YES) and DUPLEXMODE(COND) for those log streams associated with the system log. This ensures that the MVS system logger



automatically copies to staging data sets if it detects that the coupling facility is not failure-independent and a single point of failure, and is therefore vulnerable to permanent log data loss.

A connection to a log stream contains a single point of failure if the coupling facility is volatile or it resides on the same CPC as the MVS system connecting to it. For example, if you have two CPCs, CPC1 and CPC2, and CPC1 has an MVS LPAR and a coupling facility, while CPC2 has only MVS LPARs, the connections from the MVS LPAR in CPC1 to the coupling facility are failure dependent—if you lose CPC1 you lose both MVS and its local buffers and the coupling facility. On the other hand, the connections from CPC2 are failure independent, because the system logger local storage and buffers are in a physically separate CPC from the coupling facility, and you would have to lose both to lose data. With DUPLEXMODE(COND), failure dependent connections result in staging sets, while failure independent connections are not allocated staging data sets.

- If you are operating with only a single coupling facility, you should define STG\_DUPLEX(YES) and DUPLEXMODE(UNCOND) for those log streams associated with the system log.
- Define STG\_DUPLEX(YES) and DUPLEXMODE(COND) for those log streams associated with forward recovery logs. If you do not, and there is a failure which causes loss of data from the log stream, you would need to take a new image copy of the associated VSAM data sets. There would be a consequent period of time until this was complete when the data sets would not be fully protected.
- If you operate a non-volatile, stand-alone coupling facility for normal logging, with a PR/SM LPAR configured as a coupling facility acting as backup, define all log streams with STG\_DUPLEX(YES) and DUPLEXMODE(COND).
- Define each staging data set to be at least the same size as the log stream share of the coupling facility, but round the average block size up to 4K.

For example, the staging data set size corresponding to the basic coupling facility space requirement for each CICS system log stream (DFHLOG) can be calculated by the following formula:

$$\text{staging data set size} = \text{entries} * \text{rnd-avg-buf} / 4096$$

where:

$$\text{rnd-avg-buf} = \text{avgbufsize (rounded up to 4K)}$$

See “Structure size for system log usage” on page 187 for the formula to calculate no. entries.

---

## DASD-only log streams

The CICS log manager supports the DASD-only option of the MVS system logger. Individual CICS log streams can use either coupling facility log structures or DASD-only logging. Reasons for defining a log stream to use DASD-only logging include:

- You do not have a coupling facility.
- You want to preserve coupling facility space for other uses.
- You do not need to share the log stream across MVS systems. (The CICS system log can never be shared.)

See page Setting up the environment for CICS log manager for advice about defining individual log streams to use coupling facility or DASD-only logging, based on their usage.

## Defining DASD-only log streams

Use the MVS IXCMIAPU utility to define DASD-only log streams to the LOGR couple data set. The basic syntax to define a DASD-only log stream is as follows:

```
DEFINE LOGSTREAM NAME(log_stream_name)
    DASDONLY(YES)
    MAXBUFSIZE(max_bufsize)
    STG_SIZE(stg_size)
    HIGHOFFLOAD(high_offload)
    LOWOFFLOAD(low_offload)
```

For detailed information about the full range of log stream attributes, see the *OS/390 Setting Up a Sysplex* manual. Figure 16 shows example definitions for a pair of log streams associated with a DASD-only system log.

```
//DEFLOGS JOB ...
//LOGDEFN EXEC PGM=IXCMIAPU
//STEPLIB DD DSN=SYS1.MIGLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//*****
//*
/* Define DASD-only log streams for CICS system log.
/*
/* The LOWOFFLOAD and STG_SIZE values are for illustration
/* only -- substitute values appropriate for your environment.
/*
/*
//*****
//SYSIN DD *
DATA TYPE(LOGR) REPORT(NO)
DEFINE LOGSTREAM NAME(region_userid.applid.DFHLOG)
    DASDONLY(YES)
    MAXBUFSIZE(64000) STG_SIZE(3000)
    LOWOFFLOAD(40) HIGHOFFLOAD(80)
DEFINE LOGSTREAM NAME(region_userid.applid.DFHSHUNT)
    DASDONLY(YES)
    MAXBUFSIZE(64000) STG_SIZE(500)
    LOWOFFLOAD(40) HIGHOFFLOAD(80)
```

Figure 16. Example definitions of DASD-only log streams. The definitions are for the CICS primary and secondary system log streams. The value **region\_userid** is the RACF userid under which the CICS address space is running; **applid** is the CICS region's VTAM APPL name (taken from the APPLID system initialization parameter).

### Using model log streams

To avoid having to define explicitly each log stream used by each of your CICS regions, you can use model log stream definitions. Using models, log streams are defined to MVS dynamically, on their first usage. Figure 17 on page 197 shows example DASD-only model definitions for CICS primary and secondary system log streams.

```

//DEFLOGS JOB ...
//LOGDEFN EXEC PGM=IXCMIAPU
//STEPLIB DD DSN=SYS1.MIGLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//*****
//*
//* Define DASD-only model log streams for CICS system log.      *
//*                                                                *
//* The LOWOFFLOAD and STG_SIZE values are for illustration      *
//* only -- substitute values appropriate for your environment.  *
//*                                                                *
//*****
//SYSIN DD *
DATA TYPE(LOGR) REPORT(NO)
DEFINE LOGSTREAM NAME(sysname.DFHLOG.MODEL)
MODEL(YES)
DASDONLY(YES)
MAXBUFSIZE(64000)
STG_SIZE(3000)
LOWOFFLOAD(40) HIGHOFFLOAD(80)
DEFINE LOGSTREAM NAME(sysname.DFHSHUNT.MODEL)
MODEL(YES)
DASDONLY(YES)
MAXBUFSIZE(64000)
STG_SIZE(500)
LOWOFFLOAD(40) HIGHOFFLOAD(80)

```

Figure 17. Example model definitions for DASD-only system log streams. The value **sysname** is the sysid of the MVS image in which the CICS region or regions are running.

For information about the mapping of CICS journal definitions to log stream names, see the *CICS System Definition Guide*.

When using model log streams, bear in mind that, if you specify a STG\_SIZE on the model definition, all new log streams created from the model have the same-sized staging data set.

## Sizing considerations for DASD-only log streams

This section discusses how to size the following types of DASD-only log stream:

- The CICS primary and secondary system log streams
- Forward recovery logs
- User journals and autojournals.

### Sizing DFHLOG

For the CICS primary system log stream (DFHLOG), it is important to:

- **Minimize the amount of data that is offloaded to secondary storage:**

The MVS system logger begins the offload process when the high offload threshold (HIGHOFFLOAD) of the log stream is reached. The offload process consists of two steps:

1. The MVS logger physically deletes the data in the log stream that has been marked for deletion by the CICS log-tail deletion process.
2. The MVS logger calculates how much data needs to be offloaded to secondary storage, based on the difference between HIGHOFFLOAD and LOWOFFLOAD, less the amount of data that has been deleted since the last offload event.

To minimize the amount of data offloaded from the CICS primary system log, you must:

- Define a suitably-sized staging data set. For advice, see “Recommendations.”

**Note:** It is possible to alter the size of a staging data set without deleting the log stream. To do this, use the UPDATE LOGSTREAM request of the MVS IXCMIAPU utility to change the value of the STG\_SIZE parameter.

- Ensure that the log-tail deletion process is working effectively. For detailed information about the log tail deletion process, see the *CICS Recovery and Restart Guide*.

- **Avoid “staging-data-set-full” events:**

A staging-data-set-full event occurs when a log stream's staging data set becomes full before the offloading of data has completed.

For advice on monitoring and avoiding staging-data-set-full events, see the *CICS Performance Guide*.

### Sizing DFHSHUNT

It is important to size the secondary system log stream (DFHSHUNT) to avoid staging-data-set-full events. However, it is normal for some data to be offloaded from DFHSHUNT to secondary storage.

### Sizing general logs

It is important to size forward recovery logs, user journals, and autojournals to avoid staging-data-set-full events. However, because CICS does not delete data from these log streams, it is normal for data to be offloaded to secondary storage.

### Recommendations

Table 12 on page 199 summarizes how you should decide on the values for various attributes on the log stream definition, and system definition.

Table 12. How to decide on the values of attributes

Facility	Attribute	Recommendation
Primary system log stream (DFHLOG)	HIGHOFFLOAD	80
	LOWOFFLOAD	Use the formula in “Sizing your DASD log streams” on page 199 to calculate a suitable value.
	MAXBUFSIZE	64000
	STG_SIZE	Use the formula in “Sizing your DASD log streams” on page 199 to calculate a suitable value.
Secondary system log stream (DFHSHUNT)	HIGHOFFLOAD	80
	LOWOFFLOAD	40 – 60
	MAXBUFSIZE	64000
	STG_SIZE	500 (4K blocks)
General log stream	HIGHOFFLOAD	80
	LOWOFFLOAD	0
	MAXBUFSIZE	64000
	STG_SIZE	Use the formula in “Sizing your DASD log streams” on page 199 to calculate a suitable value
CICS system	AKPFREQ	4000

Table 12. How to decide on the values of attributes

Facility	Attribute	Recommendation
Primary system log stream (DFHLOG)	HIGHOFFLOAD	80
Primary system log stream (DFHLOG)	LOWOFFLOAD	Use the formula in “Sizing your DASD log streams” to calculate a suitable value.
Primary system log stream (DFHLOG)	MAXBUFSIZE	64000
Primary system log stream (DFHLOG)	STG_SIZE	Use the formula in “Sizing your DASD log streams” to calculate a suitable value.
Secondary system log stream (DFHSHUNT)	HIGHOFFLOAD	80
Secondary system log stream (DFHSHUNT)	LOWOFFLOAD	40 – 60
Secondary system log stream (DFHSHUNT)	MAXBUFSIZE	64000
Secondary system log stream (DFHSHUNT)	STG_SIZE	500 (4K blocks)
General log stream	HIGHOFFLOAD	80
General log stream	LOWOFFLOAD	0
General log stream	MAXBUFSIZE	64000
General log stream	STG_SIZE	Use the formula in “Sizing your DASD log streams” to calculate a suitable value.
CICS system	AKPFREQ	4000

**Note:** Startup may take longer than you experienced when using earlier releases. This is due in part to the allocation and formatting of the staging data sets. The increased time that startup takes is dependent on such things as:

- Size of staging data set (STG\_SIZE)
- DASD speed
- DASD contention

It can be reduced by avoiding the use of unnecessarily large staging data sets.

### Sizing your DASD log streams

If you are capacity planning for new applications, the following sections help you to calculate your space requirements.

The formulae provided help you to calculate values for:

- LOWOFFLOAD
- STG\_SIZE

You must base your calculations on the journaling requirements of your applications. This provides the starting point for the following formulae.

You can calculate **LOWOFFLOAD** for DFHLOG using the following formula:

$LOWOFFLOAD = (trandur * 90) / (akpintvl + trandur) + 10$  (where RETPD=0 specified)

or

$LOWOFFLOAD = (trandur * 90) / (akpintvl + trandur)$  (where RETPD=dddd specified)

where:

- akpintvl is the interval between activity keypoints. It can be calculated as follows:

$akpintvl = AKPFREQ / ( (N1 * R1) + (N2 * R2) + (Nn * Rn) )$

where:

- N1, N2 . . . . Nn is the transaction rate for each transaction (transactions per second).
- R1, R2 . . . . Rn is the number of log records written by each transaction.
- trandur is the execution time (between syncpoints) of the longest-running transaction that runs as part of the normal workload.

If this duration is longer than akpintvl value, you can either:

- Increase the value of AKPFREQ, so increasing the value of akpintvl (as long as this does not result in an unacceptably large staging data set size).
- Change the application logic to cause more frequent syncpoints.
- Calculate a staging data set size based on a shorter transaction duration, and accept that offloading to secondary storage occurs when the long-running transaction is used.

You can calculate **STG\_SIZE** for DFHLOG using the following formula:

Staging DS size = (AKP duration) \* No. of log writes per second  
for system log (no. of 4k blocks)  
where AKP duration = (CICS TS 390 AKPFREQ)/(No. buffers per second)

The values for the number of log writes per second and buffer puts per second can be taken from your statistics. The value for log writes per second should not exceed 30.

## Converting a DASD-only log stream to use a coupling facility

You can upgrade a DASD-only log stream to use a coupling facility structure, without having to delete and redefine the log stream. To do this:

1. Make sure that there are no connections (neither active nor failed) to the log stream.
2. Use the UPDATE LOGSTREAM request of the MVS IXCMIAPU utility. Specify the STRUCTNAME keyword, and let the DASDONLY keyword default to 'NO'.  
For example:

```

//LOGUPDT JOB ...
//LOGUPDT EXEC PGM=IXCMIAPU
//SYSPRINT DD SYSOUT=A,DCB=RECFM=FBA
//*****
//*
//* Convert DSAD-only log stream to coupling facility log stream.*
//*
//*****
//SYSIN DD *
  DATA TYPE(LOGR) REPORT(NO)
  UPDATE LOGSTREAM NAME(region_userid.applid.DFHLOG)
                    STRUCTNAME(LOG_DFHLOG_001)
                    STG_DUPLEX(YES) DUPLEXMODE(COND)

```

Figure 18. Converting a DASD-only log stream to use a coupling facility structure. This example shows the CICS primary system log stream. The value **region\_userid** is the RACF userid under which the CICS address space is running; **applid** is the CICS region's VTAM APPL name (taken from the APPLID system initialization parameter).

**Note:**

1. If you want to upgrade a DASD-only log stream to a coupling facility log stream that does *not* use a staging data set, you must explicitly specify STG\_DUPLEX(NO). (This is because the DASD-only log stream by definition uses a staging data set; unless you specify STG\_DUPLEX(NO), this is retained by the coupling facility log stream.)
2. You cannot use UPDATE LOGSTREAM to convert a log stream that uses a coupling facility structure to one that uses DASD-only. To do this, you must delete and redefine the log stream.

---

## Analyzing SMF Type 88 records

When reviewing the output from the system logger reports produced by IXGRPT1, IXGRPT1J, and IXGRPT1L, look at the following key fields for CICS system logs:

- The number of bytes deleted from primary storage should be close to the number of bytes written
- The number of bytes deleted from the system log *after* writing to offload data sets should be very low:
  - If this number is high, overhead is being incurred to move data to the offload data set only to be later deleted.
  - This is a key indicator that log tail deletion is not working as effectively as it should.
  - Check the MVS system log for any DFHRM0205 and DFHLG0743 messages from the affected CICS region.
  - Look for long running tasks (using CICS monitoring data or a system monitoring package), or check if AKPFREQ is too high.
- In general offloads are acceptable, but offloads triggered by NTRY FULL indicators are not a good sign:
  - NTRY FULL indicates that the entry to element ratio is too high
  - It is probably the result of having unlike logstreams defined in the same structure
  - The offloads are being triggered by all the entries being used rather than triggered by the HIGHOFFLOAD value.
- TYPE3 I/O counts should not appear in the statistics for coupling facility log streams, because these indicate that I/O is being initiated when over 90% of the elements for the log stream are in use.
- Average buffer size is important because:

- If over 4K, the writes are asynchronous, with a greater overhead, rather than synchronous
- Buffer size is used to determine the entry to element ratio.
- If MAXBUFSIZE specified on the log stream definition is *less than* 65532 bytes, the element size is 256 bytes.
- If MAXBUFSIZE is 65532 bytes, the element size is 512 bytes.
- The entry to element ratio is calculated as (average-buffer-size plus 4 divided by 4)

---

## Managing secondary storage

This section contains advice on how to manage secondary storage—that is, log stream data sets.

## Managing log data sets

You are recommended to use System Managed Storage (SMS) to manage log stream data sets. You can specify the SMS characteristics of log data sets in a number of ways, depending on your installation:

### Using automatic class selection (ACS) routines

You can use installation-written automatic class selection (ACS) routines to assign log data sets to SMS classes.

### Using the LOGR policy,

When you define or update a log stream definition in the LOGR policy, you can assign the SMS storage class, data class, and management class for both the DASD log data sets and staging data sets.

- Use LS\_DATACLAS to specify the SMS data class to be used for log stream data set allocation.
- Use LS\_STORCLAS to specify the SMS storage class to be used for log stream data set allocation.
- Use LS\_SIZE to specify the size, in 4K blocks, of the log stream DASD data sets. Specify a size so that each data set can contain multiple offloads of the primary storage: this is particularly important where *all* the data is offloaded for a log stream, as in the case of user journals and forward recovery logs. The MVS system logger issues message IXG256I if you specify less than 64K.

If you omit the size parameter, the size is taken from the ALLOCxx member of PARMLIB (the default is 2 tracks, which leads to a high number of new data set allocations). Specify a size that is large enough to avoid a high frequency of new data set allocations—aim for a new data set to be allocated less often than once an hour.

### SHAREOPTIONS(3,3)

Always define logger data sets with SHAREOPTIONS(3,3), whether the system is a part of a multiple-member sysplex or a monoplex. The common symptom of not having SHAREOPTIONS(3,3) is return code 84A or 403 from the logger.

For more information about managing log data sets, see the *z/OS MVS Setting Up a Sysplex* manual.



## Log tail management

Redundant data should be deleted from log streams periodically, to conserve storage, and because the MVS system logger imposes a limit on the number of data sets per log stream.

### The system log

CICS manages the system log by deleting records, for completed units of work, during activity keypoint processing (log-tail deletion). With an appropriately sized log stream, the system log data remains in primary storage, so avoiding the overhead of data spilling to DASD.

Note that:

- **The CICS system log should be used only for short-lived data required for recovery purposes.** You should not write user records for such things as audit trails to it.
- **You should allow CICS to manage the size of the system log.**

However, if historically you have used the system log for such things as audit trails, you may need to preserve system log data beyond the time it would normally be deleted by CICS. You can use the RETPD MVS parameter to preserve system log data. Define DFHLOG and DFHSHUNT to MVS with AUTODELETE(NO) and RETPD(dddd). The default values are AUTODELETE(NO) and RETPD(0.) Specifying AUTODELETE(NO) means that CICS, rather than MVS, retains control of the log-tail trimming process; dddd is the number of days for which data is to be retained. This causes the MVS logger to physically delete an entire log data set when *all* of the data in the data set:

1. Has been marked for deletion by the CICS log-tail trimming process
2. Is older than the retention period specified for the log stream.

You can view log data that has been marked for deletion by CICS but not yet physically deleted by MVS, using the DFHJUP utility program or the VIEW=ALL option of the MVS IXGBRWSE macro.

### General logs

The number of data sets per log stream recognized by the MVS logger is several million. This means that, in general, you do not need to be concerned about the limit being reached.

You can cause redundant data to be deleted from log streams automatically, after a specified period. To arrange this for general log streams, define the logs to MVS with AUTODELETE(YES) and RETPD(dddd), where dddd is the number of days for which data is to be retained. This causes the MVS system logger to delete an entire log data set when all the data in it is older than the retention period (RETPD) specified for the log stream.

**Note:** Support for the removal of the 168 data set limit, which applied only in early releases of OS/390, and support for the AUTODELETE and RETPD parameters, requires the sysplex's LOGR couple data set to have been formatted using OS/390 Release 3 or later. The removal of the 168 data set limit also requires the LOGR data set to have been formatted with DSEXTENT(nnnnn).



---

## Chapter 28. Enabling Unicode data conversion by z/OS

z/OS can provide support for the conversion of Unicode data (either UTF-8 or UTF-16) to any of the EBCDIC CCSIDs, currently supported by CICS. If you intend to make use of this capability, you must enable the z/OS conversion services and install a conversion image which specifies the conversions that you want CICS to perform. See “Selecting conversion images” for more information.

### Important

z/OS Support for Unicode must be activated before you start CICS.

Refer to the instructions in the *z/OS Support for Unicode: Using Conversion Services* manual, to set up your system and configure a suitable conversion image.

If z/OS conversion services are not enabled, a message is issued by CICS to indicate this. That message can be suppressed if you do not need these services.

- If the message is encountered when starting a CICS region that is expected to make use of these services, then an IPL is necessary to enable the z/OS conversion services.

To discover the status of z/OS conversion services after an IPL, use one of these commands:

**/D UNI** To show whether z/OS conversion services were enabled.

**/D UNI,ALL**

To show whether z/OS conversion services were enabled, and which conversions are supported by the system.

from an MVS console. For details of this, see the *z/OS Support for Unicode: Using Conversion Services* manual SA22-7649

---

## Selecting conversion images

Appendix F of the *z/OS Support for Unicode: Using Conversion Services* manual, SA22-7649, lists the supported conversions. CICS supports any of these character conversions by means of the z/OS conversion services.

The conversions entries that you select must not specify type ER.

The z/OS conversion services support big-endian byte ordered data (UTF-16BE). They do not support little-endian byte ordered data (UTF-16LE). When asked to convert UTF-16LE source data to some other CCSID, CICS transforms the data to the big-endian form of UTF-16 before calling the z/OS conversion services. When the target data is identified as UTF-16 (CCSID 1200), CICS receives the converted data from the z/OS services in big-endian byte order, and returns it in this form.

There are 3 CCSIDs for UTF data:

- CCSID 1202 indicates UTF-16LE
- CCSID 1201 indicates UTF-16BE
- CCSID 1200 indicates that the data carries a byte order marker which must be examined to see if what follows is big-endian or little-endian.

When handling CCSID 1200, CICS respects the byte order marker for inbound conversions, but is not able to retain that information when handling a related outbound conversion. All outbound data for CCSID 1200 is UTF-16BE. Application programmers need to know about this and perform their own BE to LE conversions if necessary.

---

## Chapter 29. Applying service to CICS Transaction Server for z/OS

Service material for CICS Transaction Server for z/OS is distributed as APAR fixes and PTFs. <sup>4</sup>Both types of change are called SYSMODs (SYStem MODifications).

Using SMP/E control statements, you can process SYSMODs in three stages:

1. The **RECEIVE** control statement moves the SYSMOD into the PTF temporary store (PTS) data set. This operation is reversed by the **REJECT** control statement.
2. The **APPLY** control statement moves the SYSMOD into the target libraries. This operation is reversed by the **RESTORE** control statement.  
At this point you can test the modified system.
3. The **ACCEPT** control statement moves the SYSMOD into the distribution libraries. This operation is not easily reversed.

When you are dealing with APAR fixes, you should **APPLY** the SYSMOD, but not accept it. If you later obtain a PTF that solves the problem in a different way, you may be asked to **RESTORE** (that is, remove) the APAR fix and **APPLY** the PTF instead.

When you are dealing with PTFs, you should **APPLY** the SYSMOD, then test it. Afterwards you can **ACCEPT** it.

For background information about SMP/E operations, see the *System Modification Program Extended: General Information* manual. For more detailed information, see the *System Modification Program Extended: Reference* manual.

---

### Load library secondary extents

CICS supports load library secondary extents that are created while CICS is executing. If you define libraries in the DFHRPL or dynamic LIBRARY concatenation with primary and secondary extents, and secondary extents are added while CICS is running, as a result of link-editing into the load library, the CICS loader detects the occurrence and closes then reopens the library. This means that you can introduce new versions of programs by using the CEMT NEWCOPY command, even if the new copy of the program has caused a new library extent.

However, you should not attempt to apply service to data sets that are used by executing CICS TS components.

---

### The CICS TS-supplied SMP/E procedure

There is a CICS TS-supplied procedure for applying service to the CICS and CICSplex SM components of CICS TS, called DFHSMPE. This procedure is tailored to your environment and stored in the *hlq.XDFHINST* library when you run the DFHISTAR job.

---

4. An APAR (Authorized Program Analysis Report) is raised when you and your IBM programming service representative agree that there is a CICS problem. You may then be given an APAR fix. When the problem has been analyzed, all users are sent a PTF (Program Temporary Fix) to correct the problem permanently on the current release. PTFs are incorporated into any future CICS release.

For information about how to apply corrective service with SMP/E, see the *System Modification Program Extended: User's Guide*.

Whenever you do any SMP/E processing on CICS or CICSplex SM software, and you use any of the examples quoted in the *System Modification Program Extended: User's Guide*, you should specify DFHSMPE as the name of the SMP/E procedure on the EXEC statement (that is, in place of SMPPROC, as used in the examples). The DFHSMPE procedure includes the following DD statement for supplying SMP/E control statements:

```
//SMPCNTL DD DSN=&&SETBDY,DISP=(OLD,DELETE)
//          DD DDNAME=DFHSMPIN
```

The ZNAME parameter of the DFHSMPE procedure generates a SET BDY command for the zone that is identified by the parameter. The command is stored in the temporary data set, SETBDY. The ZNAME parameter is set to the value of *zonename* that you specify for the TZONE parameter. If you do not specify any value for *zonename* for the TZONE parameter of the DFHISTAR job, *zonename* (and the ZNAME value) defaults to TZONE.

**Note:** The ZNAME parameter also generates a SET BDY command in DFHAUPL, the CICS TS procedure supplied for assembling and link-editing CICS control tables.

If you supply an **override** SMPCNTL DD statement in the job that executes DFHSMPE, remember that it must come **before** any DD statements that are additional to the procedure. Furthermore, if you provide an override, you will get the following MVS system message:

```
IEF686I DDNAME REFERRED TO ON DDNAME KEYWORD IN PRIOR STEP WAS NOT RESOLVED
```

You receive this message because the DD statement for DFHSMPIN is missing as a result of the SMPCNTL DD override. However, the message is not a JCL error, and does not prevent the step from running successfully with a return code of 0.

If you supply any SMP/E control statements in your job via the DFHSMPIN ddname, they are prefixed by a SET BDY for the zone that you specify on the ZNAME parameter. It does not matter if you are running SMP/E with a command that does not need this SET BDY statement; it does not affect the execution of your job.

---

## APAR fixes

Generally, you should **not** ACCEPT APAR fixes into distribution libraries. Subsequent PTFs may not include the APAR fix, and you may need to reapply the APAR fix.

If two APAR fixes are dependent on one another, and each is a prerequisite of the other, you must apply them both in the same SMP/E APPLY processing step.

---

## PTFs

PTFs are intended for all users to install to avoid possible problems.

A PTF may contain fixes for several different problems. This means that several APAR fixes reported in RETAIN® may all be superseded by the more permanent PTF, which:

- Provides card-image changes that are functionally equivalent to those in the APAR fix.
- Contains object-module replacements for preassembled CICS TS programs.

For further information about using SMP/E to apply service, see the *System Modification Program Extended: User's Guide*.

---

## CICS service considerations

If you use the CICS TS-supplied SMP/E usermod to install a module into the LPA (for example, into the *hlq.SDFHLPA* library), and later apply service to that module, it is the LPA-resident version of the module that is serviced. If you have not used the SMP/E usermod to install the module into the LPA, it is the original version in the *hlq.SDFHAUTH* library or *hlq.SDFHLOAD* library that is serviced.

Once you have installed CICS, and before you start the post-installation tasks described in this book, you should change the `TEMPLIB` parameter and the `SYSPROC DD` statement of the `DFHISTAR` job to refer to the *hlq.SDFHINST* library. This ensures that if you need to apply service to any of the skeleton jobs, the changes (applied to the *hlq.SDFHINST* library) are used in subsequent runs of `DFHISTAR`. In any such subsequent runs of `DFHISTAR`, you can use the `SELECT` parameter to select any jobs, affected by service, to be regenerated.

### Note:

1. If `DFHISTAR` is serviced, you should add the service changes to your `DFHISTAR` module in the *hlq.TDFHINST* library (to preserve your current installation parameters) or respecify your current installation parameters in the serviced `DFHISTAR` module (which you can copy from the *hlq.SDFHINST* library to the *hlq.TDFHINST* library).
2. Linkage editor messages `IEW0461`, `IEW2454`, `IEW2646`, `IEW2651` and `IEW2689` are produced during the `APPLY` stage for unresolved external references. These are issued, giving a return code of 4, when some CICS load modules are link-edited during PTF installation. You can ignore these `IEWxxxx` messages because they are produced for component object modules of executable CICS load modules.
3. `JCI650D` and `JCI650E` PTFs to ship Java service are often significantly larger than those for the base CICS product and may require more system resources during `APPLY` processing. To avoid errors caused by insufficient storage, it is recommended that the SMP/E `APPLY` step for such PTFs does not have a restricted region size. If a region size limit is used and the `APPLY` fails with errors relating to insufficient storage, it may be necessary to increase or remove the limit for the SMP/E job. In some cases a region size of 500M or more may be required.

---

## Servicing the CICS messages data set

Some IBM-supplied service may include changes to CICS messages, and associated changes to the CICS messages data set, `DFHCMACD`, used by the CICS-supplied transaction `CMAC`. When you have received and applied the service, you can update the CICS messages data set by running the job `DFHCMACU`. `DFHCMACU` is tailored to your CICS environment and stored in the *hlq.XDFHINST* library when you run the `DFHISTAR` job.

If a PTF contains an update to the `DFHCMACD` data set, you will see a `++HOLD` statement during the `APPLY` processing of the PTF to notify you that the

DFHMACD data set needs to be updated. The PTF will include a member called DFHxxxxx, where xxxxx is the APAR number that is associated with the PTF. You should amend the DFHMACU job so it refers to the appropriate service member of the target library *hlq.SDFHMSG*s (that is, DFHxxxxx on the SYS01 card corresponds to the DFHxxxxx part shipped by the PTF). When you submit the DFHMACU job, it updates the entries in the DFHMACD data set for all messages that are changed by the IBM supplied service.

If you are applying more than one PTF which changes the DFHMACD data set, you should either run the DFHMACU job for each PTF, or alternatively, you may include all the PTFs within one job run, by altering the DFHMACU job as follows:

```
//CMACUPD EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//AMSDUMP DD SYSOUT=*
//SYS01 DD DSN=CICSTS32.CICS.SDFHMSG(SDFHXXXXX),DISP=SHR
//SYS02 DD DSN=CICSTS32.CICS.SDFHMSG(SDFHYYYYY),DISP=SHR
.
.
//DFHMACD DD DSN=&DSINDEX.DFHMACD,DISP=SHR
//SYSIN DD *
  REPRO INFILE (SYS01) -
  REPLACE -
        OUTFILE (DFHMACD)
  REPRO INFILE (SYS02) -
  REPLACE -
        OUTFILE (DFHMACD)
.
.
/*
```



---

## Chapter 30. Applying service to CICSplex SM

This section contains information about the service material for CICSplex SM that is distributed as corrective or preventive service. Both types of changes are called system modifications (SYSMODs). SYSMODs are processed using SMP/E control statements.

For background information on SMP/E operations, see the *System Modification Program Extended: General Information* book. For more detailed information, see the *System Modification Program Extended: Reference* book. For information about how to apply corrective service using SMP/E, see the *System Modification Program Extended: User's Guide*.

---

### CICS Transaction Server for z/OS-supplied SMP/E procedure

For all CICS/ESA and CICS Transaction Server systems, the procedure for applying service is called DFHSMPE. This procedure is customized by the DFHISTAR job stored in the CICSTS32.CICS.XDFHINST library.

For full details about applying service to the CICSplex SM component of CICS TS, see Chapter 29, "Applying service to CICS Transaction Server for z/OS," on page 207.

---

### Applying PTFs to CICSplexes running CICS Transaction Server for z/OS, Version 3 Release 2 and later

Only use the procedure described below if all CMASs in your CICSplex are running CICS Transaction Server for z/OS, Version 3 Release 2 or a later release of CICSplex SM. If any CMAS in your CICSplex is running CICS TS or an earlier release of CICSplex SM, please consult the documentation that comes with the PTF for advice on how to proceed.

Some PTFs modify CICSplex SM resource tables. The documentation will tell you if this is the case. Use the procedure described below if the PTF modifies a CICSplex SM resource table:

- First apply the PTF to the maintenance point for the CICSplex.
- If the PTF modifies a repository record, upgrade the repository at the maintenance point before restarting the maintenance point CMAS.
- After the maintenance point CMAS has been restarted, apply the PTF to any local MASs connected to the maintenance point CMAS. You can restart these local MASs one at a time. There is no need to restart them all together.
- After the maintenance point CMAS has been restarted, apply the PTF to any other CMASs in the CICSplex. You can do this at the same time as the MASs connected to the maintenance point are being updated. You can update the non-maintenance point CMASs one at a time and there is no need to restart them all together.
- If the PTF modifies a repository record, upgrade the repository for the non-maintenance point CMAS before restarting it.
- Once a non-maintenance point CMAS has been restarted, you can apply the PTF to any local MASs connected to that CMAS and restart them. You can restart these local MASs one at a time. There is no need to restart them all together.

In other words, update the maintenance point CMAS first, then the other CMASs, and only update a MAS after its owning CMAS has been updated.

A PTF might contain additional documentation, giving further instructions specific to that PTF.

---

## **CICSplex SM service considerations**

If you use the CICS TS- supplied SMP/E USERMOD to install modules into the LPA (for example, into the CICSTS32. CPSM.SEYULPA library), and later apply service to that module, it is the LPA-resident version of the module that is serviced. If you have not used the SMP/E USERMOD to install the module into the LPA, it is the original version in the CICSTS32. CPSM.SEYUAUTH library or CICSTS32. CPSM.SEYULOAD library that is serviced.

After applying CICSplex SM service, ensure that all CICSplex SM regions are running with a consistent set of CICSplex SM libraries. Failure to do so may cause unpredictable results.

More information on applying service to CICSplex SM can be found in “CICS Transaction Server for z/OS-supplied SMP/E procedure” on page 211.

---

## Part 4. Getting ready to run CICS

This part describes how to tailor the CICS-supplied skeleton jobs, apply service to CICS and create the CICS data sets.



## Chapter 31. Getting ready to run CICS

This describes how to tailor the CICS TS supplied skeleton jobs, apply service to CICS and create the CICS data sets.

This part describes how to tailor the CICS TS supplied skeleton jobs, apply service to CICS TS and create the CICS TS data sets. It also describes how you can use DL/I support with CICS, how to include MRO and ISC in your CICS region, and how to use the CICS TS installation verification procedures (IVPs) to confirm that CICS TS is operational. It contains the following chapters:

### Tailoring the CICS-supplied skeleton jobs

This explains how to tailor jobs to your CICS TS environment

If you used CBPDO or the ServerPac to install CICS TS, edit and run DFHISTAR to tailor the CICS TS supplied skeleton jobs that create the CICS TS data sets and run the CICS TS IVPs.

If you have used the distribution tape to install CICS TS, as described in the *CICS Transaction Server for z/OS Program Directory*, you would normally have tailored the skeleton jobs already, and should now be able to proceed to “Creating the CICS data sets” on page 218.

Which ever method you used to install CICS TS, you can edit and run DFHISTAR several times, to create different copies of the skeleton jobs or subsequently change them. For example, to create several copies of DFHDEFDS to define data sets unique to several CICS regions, or if you have to apply service to any of the installation-related jobs. This enables you to tailor the jobs to your CICS environment after you have loaded the CICS software into the SMP/E-supported CICS libraries.

### The CICS installation libraries

When you use CBPDO to install CICS TS, you use the installation libraries shown in Figure 19.

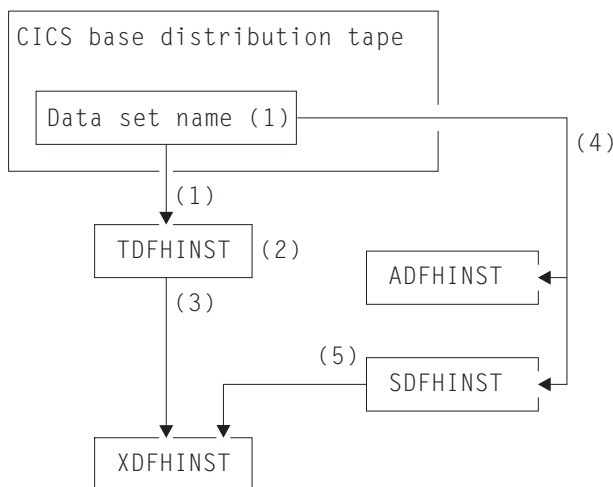


Figure 19. Installation libraries for this release

The names of the CICS installation libraries in Figure 19 on page 215 and their use is explained in the following notes which refer to the numbers in the figure:

1. Skeleton installation-related jobs are copied from data set HCI 640.F2 on the distribution tape into *hlq.TDFHINST*.

***hlq.TDFHINST***

is used to store the DFHISTAR that you edit and run to tailor the skeleton installation-related jobs to your CICS environment. Until you have installed the CICS software into the SMP/E-supported CICS libraries, this library also stores the skeleton jobs to be tailored.

2. You edit DFHISTAR in the *hlq.TDFHINST* library, to specify CICS installation parameters specific to your CICS environment.
3. When you run DFHISTAR, the tailored copies of the skeleton jobs are copied from the *hlq.TDFHINST* library to the *hlq.XDFHINST* library.

***hlq.XDFHINST***

is used to store the tailored, executable, copies of the skeleton jobs that are to be run.

4. To install CICS, you run the tailored copies of the CICS-supplied installation jobs to transfer the CICS software from the distribution tape to the *hlq.ADFHINST* and *hlq.SDFHINST* libraries.

***hlq.ADFHINST***

is the SMP/E-supported distribution installation library.

***hlq.SDFHINST***

is the SMP/E-supported target installation library. After you have installed the CICS software into this and other SMP/E-supported libraries (named SDFHxxxx and ADFHxxxx), the skeleton jobs that you should use on any later runs of DFHISTAR are stored in the SDFHINST library.

**Note:** The actual names of the TDFHINST and XDFHINST libraries, and the prefix for those and other CICS libraries, are defined in DFHISTAR, which you edit as described in this chapter.

## What you should do

To tailor the skeleton jobs you must run DFHISTAR. For information on how to do this, and the parameters involved, see *CICS Transaction Server for z/OS Program Directory*.

## Running DFHISTAR

When you have edited DFHISTAR with the values for installation parameters for your CICS environment, submit DFHISTAR.

When you run DFHISTAR, it tailors the skeleton jobs selected in the DFHISTAR input (by the SCOPE or SELECT parameter) to your environment and adds them to the library that you specified on the LIB parameter (by default, *hlq.XDFHINST*). If necessary, DFHISTAR creates the library that is specified on the LIB parameter.

A Table in section 6.4 of *Program Directory for CICS Transaction Server for z/OS* lists those skeleton jobs installed in the *hlq.SDFHINST* library that you can tailor by running DFHISTAR.

**Note:** You must specify the full name of the installation library from which the skeleton jobs are obtained, on the `TEMPLIB` parameter and `SYSPROC DD` statement of `DFHISTAR` (by default, `hlq.TDFHINST`). For the post-installation tasks that are described in this book, you should specify `TEMPLIB SDFHINST`.

`DFHISTAR` produces a job log and, if necessary, an error code:

- The output job log lists the values that were actually used for the parameters of `DFHISTAR`.
- If any error occurs when running `DFHISTAR`, an error code of 4 or 12 is returned. For error code 4, the skeleton jobs are tailored and added to the `hlq.XDFHINST` library. For error code 12, the skeleton jobs are not tailored or copied. To resolve the cause of the error, examine the output job log and, if necessary edit and submit `DFHISTAR` again.

## CICSplex SM post-installation members

`DFHISTAR` allows you to customize post-installation members, so that you can create a simple CICSplex SM configuration. The post-installation members are listed according to area: Members for a CMAS, a WUI and a managed CICS system (MAS).

A number of CICSplex SM post-installation members are delivered, as skeletons, in the `TDFHINST` library. When you run `DFHISTAR`, the post-installation members are customized and saved in the `XDFHINST` library. These members allow you to create a simple CICSplex SM configuration that consists of a CMAS, a WUI and a managed CICS system (MAS).

Post-installation members are split into three areas as shown in Table 13, Table 14 and Table 15 on page 218.

*Table 13. Post-installation members for a CMAS*

Member	Description
EYUCMASJ	JCL to start a CMAS. It executes EYUCMASP.
EYUCMASP	Starts a CMAS. This CMAS uses the CICS-supplied sample table, <code>DFHSIT6\$</code> , but appropriate override values are supplied in the job.
EYUCMS0P	EYUPARM parameters for a CMAS.
EYUCMSDS	JCL to create and initialize the data sets for a CMAS.
EYUCMSSP	CICS SIT overrides for a CMAS.

For more information on CMAS data set customization, see “CMAS data set creation and customization” on page 267.

*Table 14. Post-installation members for a WUI*

Member	Description
EYUJWREP	JCL to delete and define a WUI data repository (this function is also included in EYUWUIDS).
EYUWUI0P	EYUPARM parameters for a WUI.
EYUWUIDS	JCL to create and initialize the data sets for a WUI.
EYUWUIIN	EYUWUI parameters for a WUI.
EYUWUIJ	JCL to start a WUI. It executes EYUWUIP.

Table 14. Post-installation members for a WUI (continued)

Member	Description
EYUWUIP	Starts a WUI. This WUI uses the CICS-supplied sample system initialization table, DFHSIT6\$, but appropriate override values are supplied in the job.
EYUWUISP	CICS SIT overrides for a WUI.

For more information on WUI customization, see “WUI data set creation and customization” on page 290.

Table 15. Post-installation members for a managed CICS system (MAS)

Member	Description
EYUCSYDS	JCL to create and initialize the data sets for a managed CICS system.
EYUCSYSJ	JCL to start a managed CICS system. It executes EYUCSYSP.
EYUCSYSP	Procedure to start a managed CICS system. The MAS uses the CICS-supplied sample system initialization table, DFHSIT6\$, but appropriate override values are supplied in the job.
EYUJHIST	JCL to delete and define a pair of history data sets.
EYULMSOP	EYUPARM parameters for a managed CICS system.
EYULMSSP	CICS SIT overrides for a managed CICS system.
EYULPMOD	JCL to apply the USERMOD function, EYU\$UM01, that moves some MAS load modules to a link pack area (LPA) library.

For more information on managed CICS system customization, see “MAS data set creation and customization” on page 276.

## Creating the CICS data sets

After you have installed CICS, and applied any necessary service, you can run the DFHCOMDS, DFHDEFDS, and DFHMACI jobs to create the CICS data sets.

### Data sets

The data sets created by the jobs described in this chapter are required by the IVPs described in “Running the installation verification procedures” on page 323.

### VSAM Data sets

If you are using z/OS 1.7 or above, read Information APAR II13288.

It reports that the supplied level of DFSMS has changed the way that index controlinterval size (CISIZE) is calculated for those VSAM data sets that have index components. As a result, large default CIsizes can be expected, and this can cause open errors during CICS start-up.

This applies to both CICS data sets and your own application data sets.

### Job Function



**DFHCOMDS**

Deletes and recreates data sets common to all CICS regions.

**DFHDEFDS**

Deletes and recreates copies of data sets that are used only by one CICS region. You run a separate copy of this job to create the data sets for each CICS region.

**DFHCMACI**

Deletes and recreates the CICS messages data set, dsindex.DFHMACD, and loads it with the data from the CICS-supplied file, DFHCMACD, in the *hlq.SDFHMSG* target library.

**DFH0JCUS**

Deletes and recreates the sample applications details data set, dsindex.SAMPLE.DFHCTCUS (and its associated alternate index and path), and loads it with the data from the CICS-supplied file, DFH0DCUS, in the *hlq.ADFHAPD2* library.

**DFH0JHLP**

Deletes and recreates the sample applications help data set, dsindex.SAMPLE.DFHCTHLP, and loads it with the data from the CICS-supplied file, DFH0DHLP, in the *hlq.ADFHAPD1* library.

When you ran DFHISTAR, these jobs were tailored to your environment and stored in the library that you specified on the LIB parameter of DFHISTAR (by default, *hlq.XDFHINST*). If you have not yet run DFHISTAR, you should do so before running any of the CICS post-installation jobs.

You can generate several copies of these jobs by rerunning DFHISTAR, selecting the jobs that you want to copy. To generate new copies of these jobs, edit DFHISTAR to specify new values for the DSINFO and SELECT parameters. Only those jobs that you name by the SELECT parameter are regenerated.

## Naming conventions

There are no restrictions on the data set names you choose for CICS data sets, other than MVS constraints. In the examples in this book, *hlq* is the high-level qualifier, and the DD name is the lowest level. If you are running multiple CICS regions, and especially if you are running CICS with XRF, you can use the CICS APPLID as a third level qualifier.

You should use the *CTGI* naming convention, as in System/390 MVS Sysplex Application Migration. For example, if CICSHTH1 is the APPLID, the data set name for the CSD would be:

```
DFHCSD DD DSN=CICSTS32.CICS.CICSHTH1.DFHCSD,DISP=SHR
```

The *CTGI* naming convention is a recommended example of a naming convention that you can use for CICS 4-character names, and is based on the 4-character *CTGI* symbol, where:

- C identifies an entire CICSplex
- T identifies the type of region
- G identifies a group of regions
- I identifies iterations of regions within a group

Where names are allowed to be up to eight characters long, as for CICS APPLIDs, the general recommendation is that the letters CICS are used for the first four characters, particularly for production regions.

If the data set is shared between an active CICS region and an alternate CICS region, use the generic APPLID; but if the data set is unique to either the active or the alternate CICS region, use the specific APPLID. For information about actively and passively shared data sets, see the *CICS System Definition Guide*.

## Creating data sets common to all CICS regions, DFHCOMDS job

You can use the DFHCOMDS job to delete and recreate the following data sets common to all CICS regions:

### Name Data set

#### DFHCSD

CICS system definition.

#### SYSIN SYSIN data set.

**Note:** The CICS-supplied DFHCOMDS job creates one of each of these data sets common to all CICS regions. If you use separate copies of any of these data sets for each CICS region, you should move and edit the appropriate statements into the DFHDEFDS job. For further information about creating multiple copies of these data sets, see “Creating several copies of the DFHCSD and SYSIN data sets.”

The DFHCOMDS job comprises five job steps:

1. **DELETE** deletes the data sets.
2. **DEFCSO** defines the VSAM cluster for the CICS system definition data set, dsindex.DFHCSD, where dsindex is defined by the DSINFO parameter of DFHISTAR.
3. **INITCSD** initializes the CICS system definition data set.
4. **DEFRPIDC** defines the VSAM cluster for ONC RPC.
5. **DEFSYSIN** creates the SYSIN PDS and copies the following modules from the *hlq.SDFHSAMP* library:

DFH\$SIPA	DFH\$SIPD	DFH\$SIPT	DFH\$SIP1
DFH\$SIP2	DFH\$SIP5	DFHRCNO	DFHRCYES

### Creating several copies of the DFHCSD and SYSIN data sets

The CICS-supplied DFHCOMDS job creates one of each of the DFHCSD and SYSIN data sets common to all CICS regions. If you use separate copies of any of these data sets for each CICS region, you should:

- Move the statements that define the data set from the DFHCOMDS job to the DFHDEFDS job.
- Edit the statements in the DFHDEFDS job to specify the symbol &REGNAME for the region qualifier in the name of the data set.

You should move and edit the appropriate data set statements before you create copies of the DFHDEFDS job for each CICS region. When you run DFHISTAR to create the new copies of the DFHDEFDS job, it substitutes your values for the CICS region qualifier (&REGNAME) and index (&INDEX) into the data set names.

**For example:** If you intend using a copy of the DFHCSD data set for each CICS region, you should copy the job steps DELCSD, DEFCSD, and INITCSD from the DFHCOMDS job to the DFHDEFDS job. You should also add the symbol &REGNAME for the qualifier to the name of the DFHCSD data set to give &DSINDEX.CICS&REGNAME.DFHCSD. If you edit DFHISTAR to select the DFHDEFDS job to be copied, and specify the following DSINFO parameter:

```
DSINFO userid.CICSTS32.CICS H3P060 3390 IDA .
```

when you run the DFHDEFDS job, it creates the DFHCSD data set called userid.CICSTS32.CICS.CICSIDA.DFHCSD for the CICS region identified by the qualifier IDA. If you change the SELECT and DSINFO parameters of DFHISTAR (to specify an appropriate new job name and qualifier for another CICS region), you can create several copies of the DFHDEFDS job to create DFHCSD and SYSIN data sets for each CICS region.

## Creating data sets unique to each CICS region, DFHDEFDS job

You can use the DFHDEFDS job to delete and recreate copies of the following data sets for each CICS region.

### **Name Data set**

#### **DFHADEM**

Resource manager for enterprise beans.

#### **DFHAUXT**

Non-VSAM auxiliary trace (A).

#### **DFHBRNSF**

Bridge.

#### **DFHBUXT**

Non-VSAM auxiliary trace (B).

#### **DFHDMPA**

Non-VSAM dump (A).

#### **DFHDMPB**

Non-VSAM (B) dump.

#### **DFHEJDIR**

Bridge.

#### **DFHEJOS**

Object store.

#### **DFHGCD**

Global catalog.

#### **DFHHTML**

HTML template data set.

#### **DFHINTRA**

Intrapartition transient data.

#### **DFHLCD**

Local catalog.

#### **DFHLRQ**

Local request queue.

#### **DFHTEMP**

Temporary storage.

**FILEA** Sample program data.

**DFHDPFMB**

The debugging profiles base data set.

**DFHDPFMP**

The debugging profiles path data set.

**DFHDPFMX**

The debugging profiles alternate index data set.

Use DFHISTAR to create a copy of the DFHDEFDS job for each CICS region. Edit DFHISTAR, specifying the parameters DSINFO and SELECT, and run it once for each region.

In DFHISTAR, specify the following parameters:

- **SELECT DFHDEFDS newname** to specify the new name by which the copy of the DFHDEFDS job is to be known.
- **DSINFO** to specify the following details of the data sets for each CICS region:
  - The high-level index (*dsindex*)
  - The serial number of the volume (*volume*)
  - The unit type of the volume (*disktype*)
  - The region qualifier (*qualifier*)

The format of the data set names is:

`dsindex.CICSqualifier.dsname`

**dsindex**

is the high-level index for the data sets, specified on the DSINFO parameter of DFHISTAR. The default is *hlq*.

**qualifier**

is the region qualifier for the data sets that are used by this CICS region, specified on the DSINFO parameter of DFHISTAR. The default is no qualifier.

**dsname**

is the name of the data set being defined.

For example, the default name for the CICS local catalog is *hlq.CICS.DFHLCDC*.

The DFHDEFDS job comprises the following job steps:

1. **DELETE** any existing copies of the data sets
2. **DEFINE** defines the clusters for the data sets
3. **INITDP** initializes the debugging profiles base data set
4. **DEFAULT** defines the alternate index for the debugging profiles data set
5. **BLDDP** builds the alternate index for the debugging profiles data set
6. **INITGCD** initializes the CICS global catalog
7. **INITLCD** initializes the CICS local catalog
8. **DEFTRACE** defines the trace data sets
9. **DEFDUMP** defines the dump data sets
10. **LOADFILE** loads the sample data into the FILEA data set
11. **LOADHTML** loads HTML templates for CICS supplied transactions.

## Creating the CICS messages data set, DFHMACI job

You can use the DFHMACI job to delete and recreate the CICS messages data set DFHMACD. This data set is used by the CICS messages facility (CICS-supplied transaction CMAC).

The DFHMACI job comprises the following job steps:

1. **DELETE** deletes any existing copies of the DFHMACD data set.
2. **DEFINE** defines the VSAM cluster for the CICS message data set dsindex.DFHMACD, where dsindex is defined by the DSINFO parameter of DFHISTAR.
3. **CMACLOAD** loads the CICS message data set with data from the CICS-supplied file, DFHMACD, in the *hlq.SDFHMSG*s target library.

### Defining the DFHMACD file and associated CICS resources

You can use the CICS messages facility to provide the CICS messages and codes descriptions online. Before you can use this facility (to access the DFHMACD data set), you must define the resources needed by the facility, and make them available to your CICS region.

The file DFHMACD, managed by CICS file control, accesses the DFHMACD data set. You must create a definition for this file in the CSD. The CICS-supplied definition for the DFHMACD file and other resources that are needed by the CICS messages facility are in the CSD group DFHMAC. The CICS startup procedure, DFHSTART, has a DD statement for the DFHMACD file, but for dynamic allocation you should copy the supplied resource definition for the DFHMACD file and add the DSNNAME option.

To use the CICS messages facility in your CICS region, you must create your own CSD group lists to include the DFHMAC group for the CICS messages facility and any other groups of resources that your CICS region needs. You must specify your new group lists on the GRPLIST system initialization parameter when you start up your CICS region. If the DFHLIST of resource groups are not included in your new group lists, you must specify DFHLIST on the GRPLIST system initialization parameter as well as your group lists. For example, GRPLIST=(DFHLIST,MYLIST,CICSHT# 1), where MYLIST and CICSHT# 1 are customer-created group lists.

You should specify the DFHMAC group of resources for the CICS messages facility only in those CICS regions that need to use the facility; for example on some terminal-owning regions, but perhaps not on data-owning regions.

## Defining the sample applications data sets

CICS provides a range of samples that you can use to help develop your own applications, and test various CICS functions (for example, as an aid to verifying that CICS has installed correctly). These programs are in the *CICS 4.1 Sample Applications Guide* and the *Designing and Programming CICS Applications* book.

Before you can use some of these samples, you must create the data sets that they use, and make them available to your CICS region, as described below. You do not need to create these data sets, unless you intend using the associated sample applications.

## The CUA text level application

You can use this sample application to demonstrate BMS support for the Common User Access (CUA) interface. The application uses an action bar, with associated pull-downs, pop-ups, and help panels. The application programs demonstrate how to code COBOL programs to display, overlay, and remove CUA style windows.

**Creating the data sets:** To create the data sets that are needed by the CUA text level application, submit the following jobs: DFH0JCUS and DFH0JHLP, installed in the *hlq.XDFHINST* library.

**Making the data sets available to CICS:** You can cause CICS to dynamically allocate the files for these data sets and open them after CICS initialization by installing the sample resource definitions in the group DFH\$CTXT. If no DD statement exists for these data sets in the CICS startup job stream, the files are allocated to the data sets with DSNAMES that are specified in the resource definitions: *hlq.SAMPLE.DFHCTCUS*, *hlq.SAMPLE.DFHCTHLP*, and *hlq.SAMPLE.DFHCTAIX*, for the data sets and the alternate index. Alternatively, you can add DD statements for the data sets to your CICS startup job, which causes CICS to use the DSNAMES specified on the DD statements instead of those in the resource definitions.

For information about this sample application, see the *CICS 4.1 Sample Applications Guide* .

## The FILEA sample application programs

This comprises four sets of command-level application programs that operate on the sample VSAM file FILEA. There is one set for each of the four programming languages that are supported, (Assembler, C, COBOL, and PL/I). These programs show basic functions, such as inquire, browse, add, and update, that can serve as a framework for your own first programs. They were all written before publication of the Common User Access guidelines.

**Creating the data set:** A copy of the data set that is needed by the FILEA application is created when you submit the DFHDEFDS job, installed in the *hlq.XDFHINST* library.

**Making the data set available to CICS:** When you tailor the CICS installation-related jobs, as described in “Tailoring the CICS-supplied skeleton jobs” on page 215, a DD statement for the FILEA data set is added to the CICS IVP jobs and the DFHSTART procedure. If you want CICS to dynamically allocate the data set and open the file, you should remove the DD statement and install a FILE resource definition with an appropriate DSNAMES. (For example, as supplied in the group DFH\$FILA.)

For information about this sample application, see the *CICS 4.1 Sample Applications Guide* .

## The CICS Application Programming Primer sample application

You can use this sample application to demonstrate the design and programming of a traditional CICS application. It provides online inquiry and maintenance facilities for a sample customer credit file in a department store. The application uses VSAM files, and 3270 display and printer terminals. It was written before publication of the Common User Access guidelines, and provides similar function (without CUA support) as the CUA sample application.

**Creating the data sets:** To create the data sets that are needed by the Primer sample application, edit and submit the sample job that is shown in Figure 20 on page 226.

**Making the data sets available to CICS:** You can cause CICS to dynamically allocate the files for these data sets and open them on first reference by installing the sample resource definitions in the group DFH\$ACCT. If no DD statement exists for these data sets in the CICS startup job stream, the files are allocated to the data sets with DSNAMES that are specified in the resource definitions: *hlq.ACCTFILE* and *hlq.ACIXFILE*. Alternatively, you can add DD statements for the data sets to your CICS startup job, which causes CICS to use the DSNAMES specified on the DD statements instead of those in the resource definitions.

For information about this sample application, see the *CICS Application Programming Primer*.

```

//DEFACCTF JOB (accounting parameters),MSGCLASS=A,MSGLEVEL=(1,1),
//      CLASS=A,NOTIFY=userid
//*
//*****
//*      CICS/ESA sample jobs to define ACCT files
//*
//* This job deletes and defines the following data sets for the
//* ACCT sample described in the CICS Application Programming Primer
//*
//* STEPS:
//* . DELETE AND DEFINE
//*   - DELETE/DEFINE THE CLUSTERS FOR:
//*     . CICSTS32.CICS.ACCTFILE
//*     . CICSTS32.CICS.ACIXFILE
//*
//* THE HIGH-LEVEL-QUALIFIER(S) OF THE DATASETS: CICSTS32.CICS
//* THE VOLUME SERIAL                          CICS31
//* THE UNIT TYPE                               3390
//*
//*****
//DELETE   EXEC PGM=IDCAMS,REGION=1M
//SYSPRINT DD SYSOUT=*
//SYSIN   DD *
DELETE CICSTS32.CICS.ACCTFILE
DELETE CICSTS32.CICS.ACIXFILE
SET MAXCC=0
/*
//DEFINE   EXEC PGM=IDCAMS,REGION=1M
//SYSPRINT DD SYSOUT=*
//SYSIN   DD *
/*           */
DEFINE CLUSTER(NAME(CICSTS32.CICS.ACCTFILE)-
              KEYS(5 0)-
              INDEXED -
              RECORDSIZE(383 383)-
              REC(80)-
              SHR(2 3)-
              VOLUMES(CICS31)) -
DATA(NAME(CICSTS32.CICS.ACCTFILE.DATA)-
      UNIQUE)-
INDEX(NAME(CICSTS32.CICS.ACCTFILE.INDEX)-
      UNIQUE)
/*           */
DEFINE CLUSTER(NAME(CICSTS32.CICS.ACIXFILE)-
              KEYS(17 0)-
              INDEXED -
              RECORDSIZE(63 63)-
              REC(80)-
              SHR(2 3)-
              VOLUMES(CICS31)) -
DATA(NAME(CICSTS32.CICS.ACIXFILE.DATA)-
      UNIQUE)-
INDEX(NAME(CICSTS32.CICS.ACIXFILE.INDEX)-
      UNIQUE)
/*
//*

```

Figure 20. Example JCL to create the Primer sample data sets

---

## CICS resource definitions for CICSplex SM

This chapter describes the following:

- “Dynamic creation of CICS resource definitions” on page 227



- “Overriding the dynamically created CICS resource definitions for CICSplex SM”
- “Journalmodel considerations in a CMAS” on page 229

## Dynamic creation of CICS resource definitions

CICS can dynamically define the CICS resource definitions required to allow the CICSplex SM CMAS, MAS agent or WUI server to be started.

This includes the COLM and COVC transactions. If these transactions are invoked, or the equivalent CPSMCONN system initialization parameter is set, CICSplex SM dynamically creates further definitions as required.

The dynamically created resource definitions and their attributes can be found in the following members of the SEYUSAMP sample library:

- EYU\$CDEF contains the default resource definitions for a CMAS.
- EYU\$MDEF contains the default resource definitions for a MAS.
- EYU\$WDEF contains the default resource definitions for a WUI server.

If the dynamically created CICS resource definitions need to be changed, see “Overriding the dynamically created CICS resource definitions for CICSplex SM.”

The CSD in use needs to be appropriate for the release of CICS in use. For example, UPGRADE the CSD to the latest release and depending on the CICS release of the region, add the CICS DFHCOMPx CSD compatibility groups to the CICS group list as necessary. See the appropriate CICS Migration documentation for details.

When all regions using the CSD have been upgraded to CICSplex SM Version 3 Release 2 or later, the CICSplex SM resource definitions that were previously defined in the CSD can be removed. See 'Deleting the previous release definitions from CSD files' in the appropriate CICS Migration documentation for details.

## Overriding the dynamically created CICS resource definitions for CICSplex SM

You do not need to update your CSD to obtain the default resource definitions supplied by CICSplex SM. You are not normally expected or required to modify these definitions. However, if you do need to modify a definition, you can do so using the required definition from the EYU\$CDEF (CMAS), EYU\$MDEF (MAS), or EYU\$WDEF (WUI) samples.

Copy the required definition(s) from the appropriate sample into a DFHCSDUP job, and make the required changes, including specifying a CSD group. This CSD group should be specified in an appropriate GRPLIST for the CMAS, MAS or WUI server. The region should then be COLD or INITIAL started, as required, to process the change.

**Note:** Some CICS resource definitions are enforced. If an inappropriate definition is found, a message is produced during CICSplex SM initialization. If CICSplex SM cannot dynamically correct the error, CICSplex SM initialization may fail.

You may wish to update the CSD for CICSplex SM if CMAS journaling is required; see “CMAS journaling” on page 270.

Another reason you may wish to update the CSD for CICSplex SM is if your WUI servers require additional import or export TDQs to make copying WUI view or

menu definitions easier. The COVI and COVE definitions from EYU\$WDEF can be used as a model. For example, if the WUI servers can access the same MVS data sets:

- Copy the COVI and COVE definitions from EYU\$WDEF into a DFHCSD job.
- Rename the COVE TDQ, for example to MYEX. This TDQ definition should have the data set name specified (this data set needs to be preallocated using data set characteristics).
- Include the MYEX definition in a group that is to be used by the exporting WUI.
- Install the MYEX TDQ, once the definition has been made on the CSD using:
  - A COLD start of the exporting WUI, if the group is in the exporting WUI's group list, or
  - CEDA INSTALL, if the WUI server is up.
- Rename the COVI TDQ, for example to MYIM. This TDQ definition should have the same data set name specified on the MYEX TDQ definition.
- Include the MYIM definition in a group that is to be used by the importing WUI.
- Install the MYIM TDQ, once the definition has been made on the CSD using:
  - A COLD start of the importing WUI, if the group is in the importing WUI's group list, or
  - CEDA INSTALL , if the WUI server is up.
- Use COVC, on the exporting WUI, to export the required definitions to the MYEX TDQ. Wait for the Export to complete.
- Use COVC, on the importing WUI, to import the required definitions to the MYIM TDQ.

If the WUI servers cannot access the same data sets, the export data set must be copied from the exporting system (once the COVC Export is complete) to the importing system (before the COVC Import is run).

The following DFHCSDUP SYSIN control statements contain the default definitions, copied from EYU\$WDEF, with the modification to the default definitions, to allow the same data set to be used to export to a copy of the COVE output TD queue and the COVI input TD queue to import into another WUI. The modification is in bold type.

```
//CSDUP EXEC PGM=DFHCSDUP
//STEPLIB DD DSN=cics.index.SDFHLOAD,DISP=SHR
//DFHCSD DD DSN=cics.dfhcscd,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
*-----*
* Create TDQUEUE definitions for import/export *
*-----*
DELETE GROUP(group_name)
DEFINE TDQUEUE(MYIM) GROUP(group_name)
DESCRIPTION(CPSM WUI IMPORT DATASET)
TYPE(EXTRA) DATABUFFERS(1) DDNAME(EYUCOVI) DSNAME(h1q.wui.screens)
ERROROPTION(IGNORE) OPENTIME(DEFERRED) TYPEFILE(INPUT)
RECORDSIZE(32000) BLOCKSIZE(0) RECORDFORMAT(VARIABLE)
BLOCKFORMAT(BLOCKED) DISPOSITION(SHR)
*
DEFINE TDQUEUE(MYEX) GROUP(group_name)
DESCRIPTION(CPSM WUI EXPORT DATASET)
TYPE(EXTRA) DATABUFFERS(1) DDNAME(EYUCOVE) DSNAME(h1q.wui.screens)
ERROROPTION(IGNORE) OPENTIME(DEFERRED) TYPEFILE(OUTPUT)
RECORDSIZE(32000) BLOCKSIZE(0) RECORDFORMAT(VARIABLE)
BLOCKFORMAT(BLOCKED) DISPOSITION(SHR)
ADD GROUP(group_name) LIST(list_name)
```

where:

**STEPLIB**

Identifies:

- `cics.index.SDFHLOAD` as the CICS load library that contains the DFHCSDUP module

**DFHCSD** Identifies `cics.dfhcsd` as the CICS CSD file to be updated.

**SYSIN** The example shows the SYSIN input for import and export transient data queues

**group\_name**

Identifies the name of your group.

**list\_name**

Identifies the name of a CSD list which you include on the GRPLIST system initialization parameter for WUI server WUI A.

For information about TD queue definitions see “Transient data queue definitions” on page 301.

A return code of 4 is expected from this run of DFHCSDUP. This is because, before adding the designated group to the CSD, the job attempts to delete any group with the same name.

You can amend the supplied definitions for a CMAS and MAS in the same way. Run DFHCSDUP including the modified resource definition from the sample EYU\$CDEF (for a CMAS) or EYU\$MDEF (for a MAS) in a CSD list that you use when starting your system.

**Note:** You should not normally run user transactions in a CMAS. However, if you do choose to define your own transactions to the CMAS, you should be aware that transaction IDs used by CICSplex SM in the CMAS have no specific format. To avoid conflict between your names and those that are used by CICSplex SM, you should review the transactions that are defined in EYU\$CDEF. For a list of these transactions, see Appendix C, “Default CICS resource definitions for CICSplex SM,” on page 395.

## Journalmodel considerations in a CMAS

If the log stream names used by the CICS-supplied DFHLGMOD group are not appropriate for your environment, copy group DFHLGMOD to a new group, where you can make your amendments. Add the new group to a CSD list and ensure the list is included for your CMAS, using the SIT GRPLIST parameter

See Chapter 27, “Defining the logger environment for CICS journaling,” on page 169 for details on how to define log streams.

**Note:** Do not operate the CMAS with log streams that are defined as DUMMY. This may cause problems when recovering the CSD or CICSplex SM data repository (EYUDREP).

See “CMAS journaling” on page 270 for details about the various CMAS journaling options that you can activate.

---

## Defining DL/I support

CICS can provide DL/I database support by using the IBM IMS Database Manager . For information about appropriate Versions and Releases, see the *CICS IMS Database Control Guide*. As they become available, versions of IMS newer than those listed will also be compatible.

You can use DL/I support with CICS through:

- Database control (DBCTL)
- CICS remote DL/I support, also known as **function shipping**

This chapter describes what you do to enable a CICS region to work with remote DL/I. For information about adding system and resource definitions for use with DBCTL, see the *CICS IMS Database Control Guide*.

The IMS libraries referred to in the job streams are identified by IMS.libnam (for example IMS.PGMLIB). If you use your own naming convention for IMS libraries, please rename the IMS libraries accordingly.

CICS provides a CICS-DBCTL interface which enables DBCTL, or IMS/ESA® or IMS/ESA DM/TM, to satisfy DL/I requests that are issued from the CICS region. New users should use this method because it is simpler than local DL/I to install and provides additional function. Details of installing and using DBCTL are in the *CICS IMS Database Control Guide*.

CICS support for access to DL/I databases using the IBM Information Management System (IMS) product is included in the base product, and no specific installation is required.

For more information about storage protection, see the *CICS System Definition Guide*.

## PDIRs

A directory of program specification blocks (PDIR) is a list of program specification blocks (PSBs) that define, for DL/I, the use of databases by application programs.

Your CICS region needs a PDIR to access a database owned by a remote CICS region (remote DL/I support). Your CICS region does not need a PDIR to access a DL/I database owned by DBCTL. For information about accessing DL/I databases owned by DBCTL, see the *CICS IMS Database Control Guide*.

The modules providing remote DL/I support are automatically loaded by CICS during startup when a DL/I PSB directory is specified via the PDIR= system initialization parameter. A PDIR is mandatory for remote DL/I support, but not required for database control support.

## Adding remote DL/I support

Remote DL/I support is included in CICS Transaction Server for z/OS, and works with one of the supported levels of IMS, (see above for details of these).

Usually, you use remote DL/I support, with either MRO or ISC connections, to access databases owned by another CICS region. You can also use CICS remote DL/I support to access, through another CICS region connected to DBCTL, databases owned by DBCTL. CICS regions accessing databases owned by DBCTL

(that is, connected to DBCTL) must be running on the same MVS image as the DBCTL system. A simple overview is given in Figure 21.

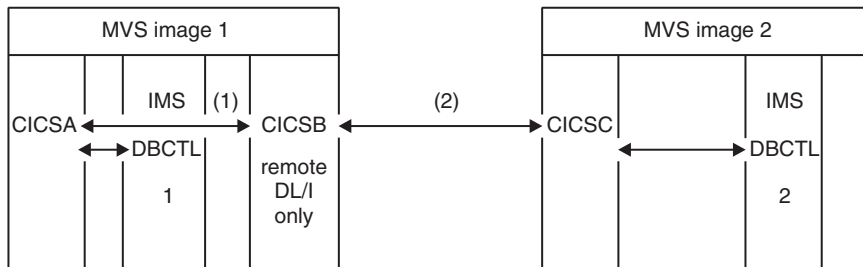


Figure 21. Using CICS remote DL/I support to access DBCTL databases

**Note:**

1. CICSB uses remote DL/I to access, through CICSA, databases owned by DBCTL 1 in MVS image 1. This is only needed if CICSB is not connected to DBCTL 1.
2. CICSB uses remote DL/I to access, through CICSC, databases owned by DBCTL 2 in MVS image 2.
3. CICSA (connected to DBCTL 1) is in the same MVS image as DBCTL 1. CICSC (connected to DBCTL 2) is in the same MVS image as DBCTL 2.

For information about accessing DL/I databases owned by DBCTL, see the *CICS IMS Database Control Guide*.

To add support in CICS for remote database access, you must:

1. Code, assemble, and link-edit a program specification blocks directory (PDIR).
2. Code the PDIR CICS system initialization parameter for remote DL/I support.

**Defining a PSB directory**

Code entries in a program specification block directory (PDIR), to indicate the identity of the remote CICS region, or regions, to which you want CICS to function ship DL/I requests. You do this by coding the SYSIDNT parameter in DFHDLPSB TYPE=ENTRY macros, which you assemble and link-edit to create a PDIR. You must also code the MXSSASZ parameter. You can, optionally, code the RMTNAME parameter to define the name by which the PSB is known in the remote CICS region. For information about creating PDIRs, see the *CICS Resource Definition Guide*.

**Coding CICS system initialization parameters for remote DL/I support**

The following is a summary of the DL/I parameters that you can, or must, code as CICS system initialization parameters:

```

PDIR={YES|xx}      SUFFIX OF PSB DIRECTORY
                   (MANDATORY for REMOTE DL/I)
PSBCHK={NO|YES}   SECURITY CHECK OF REMOTE TERMINAL INITIATING A TRANSACTION
XPSB={YES|name|NO} PSB ENTRIES TO BE CHECKED BY RACF

```

For details of these (and other) system initialization parameters, see the *CICS System Definition Guide*.

## Global user exits for DL/I

The following global user exits, if enabled, can be invoked when you have DL/I applications:

### **XDLIPRE and XDLIPOST**

These exits follow the issue of an EXEC DLI command or DL/I call; XDLIPRE before the request is processed and XDLIPOST after the request is processed. If you are running CICS with remote DL/I support, these exits are invoked in both the CICS region executing the DL/I transactions (the AOR), and the CICS region to which the DL/I requests are function shipped (the DOR). However there are restrictions on what actions can be performed by an exit program running at exit point XDLIPRE or XDLIPOST when running in a DOR.

### **XRMIIN and XRMIOU**

You can use these exits to monitor activity across the resource manager interface (RMI). For example, you can monitor control being passed to and from DFHEDP for EXEC DLI requests, DFHDBAT for DBCTL requests, or DSN2EXT1 for DB2 for DB2 commands. XRMIIN is invoked just before control is passed from the RMI to a task-related user exit. XRMIOU is invoked just after control is passed back to the RMI.

For programming information about these exits, see the *CICS Customization Guide*.

---

## Adding CICS support for programming languages

This section describes the steps necessary to add runtime support for the programming languages used with the CICS command level (EXEC) programming interface. Complete the appropriate actions described in the following before installing your application programs.

To write CICS application programs that request CICS services through the command-level application programming interface (API), you can use assembler language, C and C++, COBOL, or PL/I.

CICS provides the support needed to run application programs written in assembler language, and Language Environment provides the required support for all the other languages. The use of Language Environment is essential in CICS Transaction Server for z/OS, Version 3 Release 2. Supported compilers are listed in the *What's New* book.

The CICS Transaction Server for z/OS, Version 3 Release 2 programming guidance documentation expects that your CICS system is using the services of Language Environment, which provides a common runtime environment for IBM implementations of assembler and those high-level languages (HLLs) supported by CICS, namely COBOL, PL/I, C, and C++.

**Note:** Runtime support for OS/VS COBOL programs has been removed. OS/VS COBOL programs cannot run under CICS Transaction Server for z/OS Version 3, and must be upgraded to a supported level of COBOL.

## Installing Language Environment support

This section describes CICS support for Language Environment and what to do to install that support. The use of Language Environment is essential in CICS Transaction Server for z/OS, Version 3 Release 2.

Language Environment support is provided by run-time libraries that establish a common execution environment for application programs compiled by high-level languages. All programs compiled by a high-level language, whether by a Language Environment-conforming compiler or not, must be run under CICS-Language Environment support.

The CICS-Language Environment interface is initialized automatically when CICS can:

1. Load the Language Environment interface modules, CEECCICS, CEEPIPI, and CEECTCB, from STEPLIB.
2. Successfully call the CEECCICS module to initialize the interface.

Language Environment initialization takes place during CICS startup, when CICS issues the message DFHAP1203I *applid* Language Environment is being initialized. The CEECCICS module is loaded, followed by a partition initialization call, before the start of second phase PLT processing. If Language Environment cannot successfully complete the initialization of all languages supported by CICS, or can only initialize some of them, it issues messages to the MVS console. If Language Environment initialization fails completely, it may be because the CEECCICS module could not be loaded, or something went wrong during the loading of a particular language routine.

### Installing CICS support for Language Environment

To enable Language Environment support to be installed correctly by CICS:

- Specify enough storage for the ERDSA to run CICS and Language Environment together. They need a minimum of 3500KB. To this minimum, add an amount of storage sufficient for your own requirements.
- Ensure the CICS-Language Environment interface module, CEECCICS, and the Language Environment modules CEEPIPI and CEECTCB are installed in an APF-authorized library defined in the STEPLIB concatenation in the CICS startup JCL. You can do this by including the Language Environment SCEERUN library in an APF-authorized library in the STEPLIB concatenation of your CICS startup job (for example, in the CICSTS32.CICS.SDFHAUTH library), or in an APF-authorized library in the MVS LNKLISTnn concatenation.
- Ensure that the program resource definitions for the Language Environment language interface modules have been added to the CICS CSD. These definitions are in the CEE group.

The CEE group is added automatically to the CSD and to the grouplist DFHLIST during CICS installation, as part of the DFHCOMDS job

The definitions are also supplied as DEFINE statements in the CEECCSD and CEECCSDX members of the SCEESAMP library. You use CEECCSDX if you are using the XPLINK compiler option for C/C++ modules.

You can add the CEE group to any CICS startup group list named in the GRPLIST system initialization parameter.

- Define the Language Environment transient data destinations, CESE, and CESO (DD names CEEMSG and CEEOUT). The CICS-supplied resource definition group, in the CSD, DFHDCTG, contains entries for CESE and CESO.

For information about the attributes needed for Language Environment transient data destinations, see the *IBM Language Environment for MVS & VM Programming Guide*, SC26-4818.

- Define the Language Environment runtime libraries on the CICS STEPLIB and DFHRPL DD statements as follows:
  - Add the SCEERUN library, which contains CEECCICS and CEECTCB, and the SCEERUN2 library, which contains support that is required for the IBM Java Virtual Machine (JVM) and also support for other programming languages, to STEPLIB or to a library in the MVS LNKSTnn concatenation.
    - Both the libraries, SCEERUN and SCEERUN2, must be APF-authorized.
  - Add the SCEECICS, SCEERUN2, and SCEERUN libraries to DFHRPL.

For example:

```
//*          CICS APF-authorized libraries
//STEPLIB DD DSN=hlq.CICS.SDFHAUTH,DISP=SHR
//          DD DSN=hlq.LE.SCEERUN2,DISP=SHR
//          DD DSN=hlq.LE.SCEERUN,DISP=SHR
//*          CICS load libraries
//DFHRPL  DD DSN=hlq.CICS.SDFHLOAD,DISP=SHR
//          DD DSN=hlq.LE.SCEECICS,DISP=SHR
//          DD DSN=hlq.LE.SCEERUN2,DISP=SHR
//          DD DSN=hlq.LE.SCEERUN,DISP=SHR
```

Use only these Language Environment runtime libraries for *all* your high-level language application programs.

## Language Environment support for COBOL

Language Environment is a prerequisite for application programs written in COBOL. For information about Language Environment, see the *z/OS Language Environment Customization* manual.

Support for OS/VS COBOL programs is now withdrawn. These programs, which had runtime support in CICS Transaction Server for z/OS Version 2, cannot run under CICS Transaction Server for z/OS Version 3. OS/VS COBOL programs must be upgraded to Language Environment conforming COBOL, and recompiled against a level of COBOL compiler supported by CICS. The *CICS Application Programming Guide* provides assistance with converting OS/VS COBOL programs to Language Environment conforming COBOL.

To run COBOL application programs:

- Install support for Language Environment, ensuring that CICS can initialize the Language Environment environment during startup.
- Install resource definitions for your programs with the LANGUAGE attribute specified as LANGUAGE(COBOL), or leave the language blank.

For your application programs, CICS can create and install program resource definitions automatically, or you can create them specifically in the CSD, and install them by using the GRPLIST system initialization parameter or CEDA INSTALL command. For more information about installing program resource definitions, see the *CICS Resource Definition Guide*.

## Language Environment support for C and C++

Language Environment is a prerequisite for application programs compiled using IBM C/C++ for MVS or SAA AD/Cycle® C/370 compilers. Language Environment incorporates the run-time libraries required for both these C language compilers. For information about Language Environment, see the *z/OS Language Environment Customization* manual.



To run under CICS your C application programs:

- Install support for Language Environment, ensuring that CICS can initialize the Language Environment environment during startup.
- Install resource definitions for your programs with the LANGUAGE attribute specified as LANGUAGE(C) or leave the language blank.

For information about installing program resource definitions, see the *CICS Resource Definition Guide*.

CICS supports application programs written in C++ that:

- Are compiled using the IBM C/C++ for MVS compiler (5655-121)
- Execute with the Language Environment run-time libraries

If you use Version 3 Release 2, or later, of the C/C++ compiler to compile a C++ program, specify the CXX parameter when options are passed to the compiler, otherwise the C compiler is invoked. Do not specify CXX if a C program is to be compiled. See the *C/C++ for MVS Compiler and Run-Time Migration Guide Version 3 Release 2* for further information.

### **Language Environment support for PL/I**

Language Environment is a prerequisite for application programs compiled using IBM PL/I for MVS or SAA AD/Cycle PL/I compilers. Language Environment incorporates the run-time libraries required for both these PL/I compilers. For information about Language Environment, see the *z/OS Language Environment Customization* manual.

PL/I support is also required if you use the Web services support in CICS; in particular, it is required if you use the CICS-supplied SOAP 1.1 and SOAP 1.2 message handler programs.

To run CICS PL/I application programs :

- Install support for Language Environment, ensuring that CICS can initialize the Language Environment environment during startup.
- Install resource definitions for the programs with the LANGUAGE attribute specified as LANGUAGE(PLI) or leave blank.

For information about installing program resource definitions, see the *CICS Resource Definition Guide*.

### **Language Environment support for Java**

Language Environment is a prerequisite for Java programs that run in JVMs in CICS. However, unlike the other languages, Java programs do not require the CICS-Language Environment interface. Java programs run with Language Environment support using MVS services (not CICS services). Java programs require the Language Environment support provided by the SCEERUN and SCEERUN2 libraries only, which can either be defined in the CICS STEPLIB, or included in the MVS linklist. The SCEERUN and SCEERUN2 libraries must also be added to DFHRPL.

---

## Verifying your Java components installation

The CICS components needed to support Java application programs are included in the base product. You also need to install the IBM SDK for z/OS, Java 2 Technology Edition, on z/OS UNIX, to provide Java support. Before you begin to set up and configure Java support in your CICS system, follow the steps below to verify that the Java components are correctly installed on your system.

1. Use the supplied checklist in “Verifying Java components checklist” to ensure that all of the required Java components are installed in the correct locations on your CICS system.
2. Authorize the hlq.SDFJAUTH library. See “Authorizing the hlq.SDFJAUTH library” on page 238 for details on how to do this.

When you have verified your Java installation, use the task list in the *Setting up Java support* topic in the *Java Applications in CICS* manual to begin setting up your Java environment. You will need to:

- Set the location for the JVM profiles.
- Specify the correct version of the IBM SDK for z/OS, Java 2 Technology Edition for the CICS region.
- Ensure that CICS has the correct permissions for the Java-related resources held in z/OS UNIX.
- Verify your Java installation using sample programs.

When setup is complete, the *Using JVMs* topic in the *Java Applications in CICS* manual tells you how to start using JVMs in your CICS region.

## Verifying Java components checklist

Check that the IBM SDK for z/OS, Java 2 Technology Edition is installed, and that the CICS-supplied files have been created.

### IBM SDK for z/OS, Java 2 Technology Edition installation

Check that a 31-bit version of the IBM SDK for z/OS, Java 2 Technology Edition is installed in z/OS UNIX. CICS Transaction Server for z/OS, Version 3 Release 2 can use either Version 1.4.2 or Version 5 of the IBM SDK for z/OS, Java 2 Technology Edition for Java support. The two versions of the SDK can co-exist on the same z/OS system, although a CICS region can only use one Java version at a time. CICS TS V3.2 supports only the 31-bit version of each SDK, not the 64-bit version.

The JAVADIR parameter of the DFHISTAR installation job for CICS specifies the install location for the IBM SDK for z/OS. This parameter is used to customize the CICS-supplied sample JVM profiles, JVM properties files and Java 2 security policy, so that they point to the SDK installation. The value in the JVM profiles determines which version of the SDK is used for Java support in a CICS region. You can switch versions at any time by changing this value after you have installed CICS.

The default for the JAVADIR parameter is java142/J1.4, which specifies the default install location for Version 1.4.2 of the IBM SDK for z/OS, Java 2 Technology Edition. For Version 5, the default install location is the java/J5.0 directory.

## CICS-supplied files

The following components are unloaded from the distribution tapes during the standard installation process described in the CICS Transaction Server for z/OS, Version 3 Release 2 Program Directory. Confirm that they are all present in the specified locations.

Note that z/OS UNIX System Services must be active in **full function** mode during the installation process to enable files to be stored in its file system.

In all of the directory paths shown here, `cicsts32` is a user-defined value, specified on the `USSDIR` parameter in the `DFHISTAR` installation job, which determines the install directory for CICS files on z/OS UNIX. This value is passed to the `uss_path` variable used by the `DFHIJVMJ` job, which creates the customized sample JVM profiles, JVM properties files and Java 2 security policy as z/OS UNIX files in the directories listed here.

Table 16. Java components checklist

Java component	Location	Comments
Directory containing CICS-supplied JAR files	z/OS UNIX directory:  <code>/pathprefix/usr/lpp/cicsts/cicsts32/lib</code>	This directory is on the base library path and classpath for all JVMs in CICS.
Sample programs	z/OS UNIX directory:  <code>/pathprefix/usr/lpp/cicsts/cicsts32/samples</code>	Demonstrate the use of IIOP, EJBs, output redirection, the JCICS classes, connectors, JDBC, pipelines and Web services.
JVM profiles  DFHJVMPR DFHJVMPD DFHJVMPG DFHJVMPH DFHJVMPJ DFHJVMPK DFHJVMPN DFHJVMPQ DFHJVMPR DFHJVMPD (DFHJVMPD is reserved for the use of CICS)	z/OS UNIX directory:  <code>/pathprefix/usr/lpp/cicsts/cicsts32/JVMProfiles</code> directory	JVM profiles specify options used in creating JVMs. These sample JVM profiles are customized for your system during the installation process. You can edit them using any text editor. <code>DFHJVMPR</code> and <code>DFHJVMPD</code> , and their associated JVM properties files, must always be available to CICS.
JVM properties files  dfjvmp.r.props dfjvmpd.r.props dfjvmpg.r.props dfjvmpj.r.props dfjvmpk.r.props dfjvmpn.r.props	z/OS UNIX directory:  <code>/pathprefix/usr/lpp/cicsts/cicsts32/props/</code>	JVM properties files are referenced by JVM profiles to specify system properties for JVMs. These samples each support a corresponding JVM profile. They are also customized during the install process. They are referenced by their full path name, for example:  <code>JVMPROPS=/pathprefix /usr/lpp/cicsts/ cicsts32/props/dfjvmp.r.props</code>

Table 16. Java components checklist (continued)

Java 2 security policy file dfjebpl.policy	z/OS UNIX directory:  /pathprefix/usr/lpp/cicsts/cicsts32/lib/security/dfjebpl.policy	A sample Java 2 security policy. It defines security properties that are suitable for JVMs that are used by enterprise beans. It is customized during the installation process. See the <i>Java Applications in CICS</i> manual for more information.
hlq.SDFJAUTH library	MVS PDSE libraries	Contains components of the SJ domain. See “Authorizing the hlq.SDFJAUTH library” for more information.

For all z/OS UNIX files, including JVM profiles, JVM properties files, and the sample security policy, case is important. CICS **does not** automatically convert the name of these files to upper case. When you use the name of a JVM profile, JVM properties file, or security policy, anywhere in CICS, you must enter it using the same combination of upper and lower case characters that is present in the z/OS UNIX file name.

## Authorizing the hlq.SDFJAUTH library

This library is the partitioned data set extended (PDSE) version of SDFHAUTH, and it contains some of the components of the SJ domain. The SDFJAUTH library is required for Java support.

A separate library is needed because these components are now built using XPLink (Extra Performance Linkage).

As for the SDFHAUTH library, you need to:

1. APF-authorize the SDFJAUTH library by adding it to the list of APF-authorized libraries in the appropriate PROGxx (or IEAAPFxx) member in SYS1.PARMLIB.
2. Provide a STEPLIB DD statement for the hlq.SDFJAUTH library in your startup job stream.

The procedure for authorizing the SDFHAUTH library is described in Chapter 7, “Authorizing the CICS and CICSplex SM libraries,” on page 79. Follow the same procedure to authorize the SDFJAUTH library.

## Amending parameters in SYS1.PARMLIB(BPXPRMxx)

The default options shipped in the BPXPRMxx members of SYS1.PARMLIB are not sufficient for use with the JVM used in CICS, which is supplied by the IBM SDK for z/OS, Java 2 Technology Edition.

In particular, the values for IPCSEMNSEMS, MAXPROCSYS, MAXPROCUSER, MAXUIDS, MAXASSIZE, MAXFILEPROC, MAXPTYs, MAXTHREADS, MAXTHREADTASKS, and MAXCPUTIME are too low for CICS usage - increase them.

### Recommendation:

Start by using the following values. Tailor them to your system's needs as you gain experience:

#### **IPCSEMNSEMS**

1000

**MAXPROCSYS**  
500  
**MAXPROCUSER**  
512  
**MAXUIDS**  
500  
**MAXASSIZE**  
2000000000  
**MAXFILEPROC**  
512  
**MAXPTYs**  
256  
**MAXTHREADS**  
10000  
**MAXTHREADTASKS**  
5000  
**MAXCPU TIME**  
2147483647

For full details, see *z/OS UNIX System Services Planning*, GA22-7800, in section 14.8, “Customizing the BPXPRMxx parmlib members”, and *z/OS MVS Initialization and Tuning Reference*, SA22-7592, in section 9.0, “BPXPRMxx (z/OS UNIX System Services parameters)”.

---

## Installing MRO and ISC support

This chapter describes what you have to do to include the following communication facilities in your CICS region:

- Multiregion operation (MRO): see “Installing MRO support.”
- Intersystem communication (ISC): see “Intersystem communication over SNA” on page 242 and “IP interconnectivity (IPIC)” on page 242.

**Note:** The information about ACF/VTAM and z/OS that is given in this chapter is for guidance only. Always consult the current ACF/VTAM or z/OS publications for the latest information.

## Installing MRO support

CICS multiregion operation (MRO) enables CICS regions that are running in the same z/OS image, or in the same z/OS sysplex, to communicate with each other. MRO does not support communication between a CICS system and a non-CICS system such as IMS.

The external CICS interface (EXCI) uses a specialized form of MRO link to support DCE remote procedure calls to CICS programs, and communication between z/OS batch programs and CICS .

MRO does not require ACF/VTAM or SNA networking facilities. The support within CICS that enables region-to-region communication is called **interregion communication (IRC)**. IRC is implemented in three ways:

1. Through support in CICS terminal control management modules and by use of a CICS-supplied interregion program, DFHIRP, loaded in the z/OS link pack area. DFHIRP is invoked by a type 3 supervisory call (SVC).

2. By z/OS cross-memory services, which you can select as an alternative to the CICS type 3 SVC mechanism. Here, DFHIRP only opens and closes the interregion links.
3. By the cross-system coupling facility (XCF) of z/OS. XCF/MRO is required for links between CICS regions in different z/OS images of an z/OS sysplex. CICS selects XCF/MRO dynamically for such links, if available.

For information about the design and implementation of interregion communication, and about the benefits of cross-system MRO, see the *Intercommunication concepts and facilities* topic in the *CICS Intercommunication Guide*.

To install support for MRO, complete the following steps (outlined in more detail in this chapter):

1. Define CICS as an z/OS subsystem.
2. Install the current versions of the DFHIRP and DFHCSVC modules in the LPA.
3. If you give the SVC a new number, and you have CICS Version 1 or Version 2 regions that use MRO, regenerate the CICS modules DFHCRC and DFHDRPA for those CICS versions, specifying the SVC number.
4. Specify appropriate system initialization parameters to enable MRO for each CICS region startup.  
If you intend using cross-system MRO (XCF/MRO) you must also:
  5. Install the required sysplex hardware and software.
  6. Define the z/OS images as systems in an XCF sysplex.  
To use the MRO support, you must also:
7. Define and install the MRO connections appropriate to your CICS environment.

Provided you complete the above steps, you can use MRO to communicate with all levels of CICS from CICS/ESA Version 4.1 onwards.

Should MRO be used to communicate between different releases of CICS, the function provided on any connection is that of the lower-level release.

### Defining CICS as an z/OS subsystem

Multiregion operation with CICS requires z/OS Subsystem Interface (SSI) support, and to obtain this you must define CICS as an operating system subsystem, as described in Chapter 11, “Defining CICS as an MVS subsystem,” on page 99.

### Installing the modules DFHIRP and DFHCSVC in the LPA

To enable your regions to communicate by MRO, you must:

1. Install the current versions of the DFHIRP and DFHCSVC modules into the LPA, as described in Chapter 15, “Installing CICS modules in the MVS link pack area,” on page 129.

**Note:** If you are running CICS with MRO at different release levels, all communicating CICS regions must use the latest DFHIRP module and the latest SVC module, DFHCSVC, on the same z/OS image. Do not use the dynamic LPA function to replace DFHIRP for migration between releases, as this can cause incompatibility between control blocks, resulting in abend situations.

2. Define the SVC module, DFHCSVC, to z/OS, as described in Chapter 12, “Installing the CICS Type 3 SVC,” on page 111.

## Installing required hardware and software for XCF/MRO

To be able to use the cross-systems MRO to communicate between CICS regions on different z/OS images, those z/OS images must be running with appropriate hardware and software. The hardware and software that are required for z/OS systems in a sysplex are in the *CICS Transaction Server for z/OS Program Directory*.

For detailed guidance information about setting up XCF/MRO, see the *CICS Intercommunication Guide*.

## Defining z/OS images as systems in an XCF sysplex

To use XCF/MRO, all participating z/OS images must be defined as part of the same sysplex, as in Chapter 22, “MVS cross-system MRO definitions,” on page 155.

**Note:** Within a parallel sysplex, where MRO communication between z/OS images is by XCF/MRO, the DFHIRP programs installed in the different z/OS images can be at different release levels. However, DFHIRP must be installed from the highest release of CICS running in an z/OS image. For example, a CICS TS for z/OS, Version 2.3 DFHIRP can communicate with a DFHIRP across XCF/MRO, but the CICS regions running in the z/OS with the CICS TS for z/OS, Version 2.3 DFHIRP cannot be later than CICS TS for z/OS, Version 2.3.

**Defining MRO connections:** Before you can use MRO, you must define and install connections with attributes appropriate to your CICS environment. For guidance information about defining MRO connections, see the *CICS Intercommunication Guide*. For reference information about CONNECTION definitions, see the *CICS Resource Definition Guide*.

## Enabling MRO for CICS startup

For each CICS region that is to use MRO, you must specify `ISC=YES` to include the intersystem communication program DFHISP.

If you want a CICS region to establish MRO communication during startup, you should also specify `YES` on the `IRCSTRT` system initialization parameter.

Alternatively, once your CICS region is running, you can establish MRO communication by using the `CEMT SET IRC OPEN` command or the `EXEC CICS SET IRC OPENSTATUS` command.

Either method establishes MRO communication with every CICS region that is:

1. Currently active.
2. Defined to your region by CONNECTION and SESSIONS definitions that are installed from the CSD. To establish MRO communication between two CICS regions, the installed CONNECTION definition must specify `INSERVICE(YES)` in both regions.

## Adding communications support between systems

This section outlines how to include communications to other systems in a CICS region.

MRO can be used only for CICS-to-CICS connections within the same z/OS image or z/OS sysplex. If you need to connect CICS to both CICS and non-CICS systems, and the remote systems could reside inside or outside the local z/OS sysplex, CICS offers the following methods:

- intersystem communication over SNA (ISC over SNA)
- IP interconnectivity (IPIC)

Unlike MRO, there are no special z/OS operating system requirements for these methods of communication.

These are described in the *CICS Intercommunication Guide*.

### **Intersystem communication over SNA**

To provide the necessary protocols to support communication between CICS regions that are in different z/OS images, or in different z/OS sysplexes, ISC over SNA uses the ACF/VTAM access method. (You can also use ISC over SNA in the same CPC, through the application-to-application facilities of ACF/VTAM.)

You must include the following management programs in your CICS regions, (by specifying the system initialization parameters that are given in parentheses):

- DFHISC – the intersystem communication program (ISC=YES).
- DFHTCP – the terminal control program (TCP=YES is the default).

**Establishing ISC over SNA:** Intersystem communication over SNA requires VTAM support, and you must specify VTAM=YES as a system initialization parameter. If VTAM is running during CICS initialization, CICS opens the VTAM ACB. If VTAM is started after CICS, opening the VTAM ACB fails, and you must open it using the CEMT SET VTAM OPEN command when VTAM is available. CICS regions cannot communicate until they have established the VTAM connection.

**Defining ISC over SNA connections:** Before you can use ISC over SNA, you must define and install connections with attributes appropriate to your CICS (and VTAM) environment. For guidance information about defining ISC over SNA connections, see the *CICS Intercommunication Guide*.

For reference information about CONNECTION definitions, see the *CICS Resource Definition Guide*.

If you intend using APPC for your ISC communications, you can take advantage of the autoinstall for APPC connections function. For information about using the autoinstall for APPC connections function, see the *CICS Resource Definition Guide*.

### **IP interconnectivity (IPIC)**

To enable support for IP interconnectivity (IPIC), you need to activate CICS TCP/IP services, and define and install some resource definitions.

### **System initialization parameter for IPIC**

To activate IPIC at CICS startup, specify TCPIP=YES and ISC=YES as system initialization parameters. The default value of the **TCPIP** and **ISC** parameters is NO. For reference information about the **TCPIP** and **ISC** system initialization parameters, see the *CICS System Definition Guide*.



## Resource definitions for IPIC

To enable IPIC:

- Define and install IPCONN definitions with attributes appropriate to your CICS environment. For guidance information about defining IPCONN definitions, see the *CICS Intercommunication Guide*. For reference information about IPCONN definitions, see the *CICS Resource Definition Guide*. IPIC connections can be autoinstalled; see the *CICS Customization Guide*.
- Define and install a TCPIP SERVICE definition with the PROTOCOL attribute set to IPIC. See the *CICS Resource Definition Guide* for reference information about TCPIP SERVICE definitions.

---

## Enabling TCP/IP in a CICS region

TCP/IP support is provided by the CICS sockets domain, with network services supplied by z/OS. The sockets domain provides support for:

### The listener

The listener monitors specified TCP/IP ports for incoming requests. It is configured by a TCPIP SERVICE resource definition to listen on a specific TCP/IP port and to attach a specified request receiver transaction to handle each connection. Once the connection has been established between a client program and a particular request receiver, all subsequent requests from the client program over that connection flow to the same request receiver. The listener supports user applications initiated by TCP/IP services for the following protocols:

### External Call Interface (ECI)

The External Call Interface (ECI) allows client applications to use TCP/IP directly to CICS regions without any intervening products such as protocol-mapping software. The External Call Interface (ECI) is supported, but not the External Presentation Interface (EPI).

See the *CICS Family: Communicating from CICS on zSeries* manual for information about the External Call Interface.

### Hypertext Transfer Protocol (HTTP)

HTTP messages are received and sent over the Internet, using CICS Web support. See the *CICS Internet Guide* for information about the transmission of HTTP messages on the Web.

### Internet InterORB Protocol (IIOP)

IIOP messages are sent between client and server applications that conform to the Common Object request Broker Architecture (CORBA). See the *Java Applications in CICS* manual for information about IIOP messages.

### IP interconnectivity protocol (IPIC)

The IPIC protocol is used for distributed program link (DPL) calls over IP interconnectivity (IPIC) connections (“IPCONN”).

### Outbound socket support

This allows CICS to initiate an IPCONN. Sockets can be created by one task, shared by other tasks, and remain active after task termination, for re-use by another task.

## Using TCP/IP in a CICS region

To use TCP/IP in a CICS region, you must provide the following:

- Communications Server must be installed in the system ( In earlier OS/390 releases this is called *eNetwork Communications Server*, or *Secureway Communications Server*). Ports belonging to Communications Server must be made available for use by the CICS region involved.
- The CICS system initialization parameter TCPIP must be set to YES.
- TCPIPSERVICE resource definitions must be provided to define each active port and the type of service associated with it. The CICS TCP/IP listener is activated for the specified ports when the TCPIPSERVICE is installed, if TCPIP (YES) has also been specified.

For information about creating TCPIPSERVICE definitions for use with IP connections, see the *CICS Intercommunication Guide*.

For reference information about TCPIPSERVICE definitions, see the *CICS Resource Definition Guide*.

- If Secure Sockets Layer (SSL) authentication is used, you must define the KEYRING system initialization parameter, to identify the RACF key ring containing the keys and X.509 certificates used in the SSL handshake.
- When using DNS connection optimization, the CICS listener regions must be configured to talk to the same name server that the z/OS Workload Manager is configured to use. This means that you may need to reconfigure the DNS server that CICS uses to resolve host names, because CICS needs to resolve its own host name (using a call to the *gethostbyaddr* function) using the DNS server configured for connection optimization in the sysplex. This may not be the system-configured name server if the sysplex is already configured for TCP/IP operation. The system name server may not even be on z/OS or on any of the systems in the sysplex.

You can change the resolver configuration of CICS either by altering system TCP/IP configuration files, or by adding or changing the DD name SYSTCPD in the CICS start-up JCL. This sets the RESOLVER\_CONFIG environment variable to the MVS dataset you have specified. This file is described in the *z/OS IBM Communications Server: IP Configuration Guide*, SC31-8775. It contains a reference to the DNS server's IP address.

You must specify at least the following:

```
NSINTERADDR n.n.n.n
```

where *n.n.n.n* is the dotted decimal address of the name server to be used.

If the DD name is not included in the startup JCL, a number of system files are searched until one is found.

**Note:** The TCPIPSERVICE resource definitions are for use only with the CICS-provided TCP/IP services, and have nothing to do with the z/OS Communications Server IP CICS Sockets interface. The TCP/IP Socket Interface for CICS is supplied with z/OS Communications Server, which is an integral part of z/OS and does not use the CICS SO domain.

A TCPIPSERVICE only supports either one specific IP address or all IP addresses (INADDR\_ANY), therefore if 2 CICS regions are required to listen on the same port at the same IP address you need to use TCP/IP port sharing.

---

## Enabling REXX for CICS

The REXX Development System for CICS and the REXX Runtime Facility for CICS--two program products collectively referred to as REXX for CICS--provide improved productivity for a wide range of CICS activities.

REXX for CICS enables you to write and execute REXX programs in a CICS region. These programs have access to most EXEC CICS commands, the CICS CEDA and CEMT transactions, and DB2 databases through the EXEC SQL interface.

To enable your CICS system to provide the facilities of REXX for CICS, perform the following steps:

- Step 1-- Consider your customization needs for REXX, see “Customization information for REXX” on page 247.

You might prefer to familiarize yourself with the procedure at the first reading by ignoring customization. (We intentionally show this step at the beginning and again at the end of the list to cater for those working through the list who need it first, and those reading it for the first time who would benefit from covering the topic last.)

- Step 2--“Step 1--Modify your RDO definitions to add required entries.”
- Step 3--“Step 2--Create the RFS filepools.”
- Step 4--“Step 3--BIND the CICS SQL program to your DB2 plan” on page 246.
- Step 5--“Step 4--Add DD statements to your CICS startup job” on page 246.
- Step 6--“Step 5--Modify member CICSTART” on page 246.
- Step 7--“Step 6--Format the RFS filepools” on page 246.
- Step 8--“Step 7--Verifying the installation” on page 247.
- Step 9--“Step 8--Accessing the supplied softcopy documentation” on page 247.
- Step 10--“Customization information for REXX” on page 247 (if you choose to skip step 1).

### Step 1--Modify your RDO definitions to add required entries

Job CICRDOR (for Runtime Facility) or job CICRDOD (for Development System) in data set CICSTS32.REXX.SCICJCL adds the entries needed by the product, including REXX/CICS profiles, VSAM files, programs, transactions, and transient data queues. The transient data queues are used for REXX/CICS IMPORT and EXPORT commands. The jobs also contains the definitions for the REXX/CICS SQL interface that authorize the transactions to the DB2 plan. Review “Changing supplied CICS transaction codes” on page 248 if you plan to modify the transaction IDs and also review “RFS filepool definitions” on page 249 if you plan to change the REXX file system (RFS) pool names or the number of pools to install. Edit the JCL (ensuring that you uncomment the entries as explained in comments at the beginning of the JCL) and run the job.

A return code of 4 is acceptable.

### Step 2--Create the RFS filepools

Job CICVSAM in data set CICSTS32.REXX.SCICJCL defines the clusters needed to create the VSAM data sets needed for the RFS filepools. See “RFS filepool definitions” on page 249 for customizing information about RFS filepools.

You receive a condition code of 8 for the delete control statements if the VSAM data sets do not exist. You receive a condition code of 0 for the define cluster control statements if the job runs correctly.

### Step 3--BIND the CICSQL program to your DB2 plan

Job CICBIND in data set CICSTS32.REXX.SCICJCL binds CICSQL to the correct DB2 plan. Edit and run the job.

You might receive condition code 4 for the job depending upon the level of DB2 being used.

### Step 4--Add DD statements to your CICS startup job

You need to add the following DD statements to your CICS startup job. See "Special exec data sets used by the REXX Development System" on page 250 for more information.

```
//CICAUTH DD DSN=CICSTS32.REXX.SCICCMDS,DISP=SHR
//CICEXEC DD DSN=CICSTS32.REXX.SCICEXEC,DISP=SHR
//CICUSER DD DSN=CICSTS32.REXX.SCICUSER,DISP=SHR
```

A DD statement for the REXX data sets must also be added to the DFHRPL concatenation.

```
//DFHRPL DD DSN=CICSTS32.REXX.SCICLOAD,DISP=SHR
```

### Step 5--Modify member CICSTART

Modify member CICSTART in data set CICSTS32.REXX.SCICEXEC. See "Customization information for REXX" on page 247 for more information.

### Step 6--Format the RFS filepools

Bring up the CICS region and sign onto the CICS region with a userid defined as an authorized user. Enter REXX, the default transaction id associated with the CICRXTY exec. You should see the following line at the top of the screen, "Enter a REXX command or EXIT to quit" and a "READ" in the lower right hand corner. The cursor is in the lower left hand corner. You have now entered the supplied exec which allows the execution of REXX and REXX/CICS commands interactively.

You may now prepare the filepools for use by entering the command: 'FILEPOOL FORMAT *pool1*' where *pool1* should be substituted by the filepool name you specified in the CICSTART exec.

**Note:** The command should be entered as shown, including the apostrophes.

The interactive environment will echo each command at the next available line on the screen and any requested output will also be displayed. The FILEPOOL FORMAT command does not display any information. To determine whether the FILEPOOL FORMAT command worked successfully, enter "SAY RC". If a "0" is displayed on the next available line, the FILEPOOL FORMAT command was successful.

Continue this process until all RFS filepools have been formatted. You only have to format the filepool when a new filepool has been defined, or if you delete and redefine the clusters for an existing filepool.

If, in the process of formatting the filepools or interactively executing REXX or REXX/CICS commands and instructions, you fill the screen, a "MORE" indicator will appear at the bottom right hand corner. To clear the screen, press the ENTER key.

Any time you want to clear the screen of data, you may press the CLEAR key. To exit from the interactive environment, you may press the PF3 key which simulates the entering of the "EXIT" REXX instruction. You may also enter the "EXIT" instruction yourself.

The interactive environment also provides for recalling previously entered commands. This is done by pressing the RETRIEVE key. The system has a default setting for this key of PF12. This may be customized using the SETSYS RETRIEVE command. Pressing the RETRIEVE key causes the previously entered line to be re-displayed at the input location. You can then modify this area if required and re-execute the instruction by pressing ENTER. Pressing the RETRIEVE key multiple times will continue to bring the next previously entered command to the input area.

## Step 7--Verifying the installation

To verify the installation has been successful, three execs have been supplied. From the interactive REXX environment, enter, CALL CICIVP1. The exec will indicate what should occur.

## Step 8--Accessing the supplied softcopy documentation

The REXX Development System for CICS/ESA and the REXX Runtime Facility for CICS/ESA Guide and Reference manual is included on the distribution tape in three different softcopy formats.

The supplied data set CICSTS32.REXX.SCICDOC contains two members.

Member CICR3270 contains the manual in LIST3270 format, a format that has an 80-character record length. CICR3270 is used as input by the online help facility which is described in "Online HELP facility" on page 251.

Member CICR3820 contains the manual in LIST3820 format. Job CICBPRNT in data set CICSTS32.REXX.SCICJCL contains a sample job, which may be modified and executed, that prints the manual to a device which supports LIST3820 formatted data.

The supplied data set CICSTS32.REXX.SCICBOOK contains one member, CICRBOOK. It contains the manual in BookManager/read format. If you have BookManager<sup>®</sup> installed you may use this data set to view the manual.

## Customization information for REXX

This section provides customizing information. It should be reviewed prior to performing the steps listed in "Step 1--Modify your RDO definitions to add required entries" on page 245. It covers the following topics:

- "Changing supplied CICS transaction codes" on page 248.
- "RFS filepool definitions" on page 249.
- "TD queues needed for IMPORT and EXPORT commands" on page 249.
- "SQL definitions used for authorizing transactions to use DB2" on page 249.
- "Special exec data sets used by the REXX Development System" on page 250.
- "Special USER ids and their usage" on page 250.
- "Other considerations" on page 251.
- "Online HELP facility" on page 251.

## Changing supplied CICS transaction codes

There are three transaction ids supplied by the product. They are: REXX, EDIT, and FLST. You can change these supplied transaction ids.

**The functions of REXX, EDIT, and FLST:** The functions of REXX, EDIT, and FLST are:

**REXX** is the default transaction id.

- If no additional operands are supplied the CICRXTRY exec starts. CICRXTRY allows the user to interactively enter REXX instructions and execute them.
- If REXX is entered and is followed by a string (separated from REXX by blanks), the string is interpreted as a REXX exec name followed by operands that are passed to the named REXX exec. This action causes the named exec to execute.

When the REXX exec ends, control is returned to CICS.

**EDIT** is the transaction id associated with the REXX Development System editor.

- If no additional operands are supplied the CICEDIT exec starts and the file "NONAME" in the user's current RFS directory is opened for editing.
- If EDIT is entered with an additional operand separated from the transaction id with a blank, the operand is interpreted as the name of a particular file in the user's current directory which is to be opened for editing.

When the EDIT session ends, control is returned to CICS.

**FLST** is the transaction id associated with the REXX Development System file list exec, CICFLST.

- If no additional operands are supplied, the CICFLST exec starts and the contents of the user's current RFS directory are displayed.
- If FLST is entered with an additional operand separated from the transaction id with a blank, the operand is interpreted as the name of a particular RFS directory whose contents are to be listed.

When the FLST session ends, control is returned to CICS.

**Changing the supplied transaction ids:** You can change the names of the supplied transaction ids, and you can add additional transactions which will call your own execs.

- The DEFTRNID commands in the member, CICSTART, in data set CICSTS32.REXX.SCICEXEC define the supplied transaction ids REXX, EDIT, and FLST, and associate them with their execs.

If you choose to change the supplied entries make sure that you update the RDO definitions to match your changes. If you do not want users to call the editor or file list execs directly from CICS, you can delete the DEFTRNID commands, for either or both of them, from CICSTART and also from the RDO definitions. Users will not be allowed access to these directly from CICS. Ensure that you do not remove the DEFTRNID statement for the CICRXTRY exec.

- If you want to add additional transactions which call your own execs directly from CICS, add RDO definitions for the transaction ids and add further DEFTRNID commands to your CICSTART exec. Your newly defined transactions become available to your users when you restart your CICS system. An authorized user can enter the DEFTRNID command directly to give immediate availability, but until the CICSTART member is changed, these definitions are lost when CICS is restarted.

## RFS filepool definitions

The supplied member, CICVSAM in CICSTS32.REXX.SCICJCL, creates the VSAM data sets for two RFS filepools. The names for these VSAM data sets can be changed to match your installation standards. If you do change these names make sure to make matching changes to the member, CICRDOD, as well. Since the RDO definitions supplied contain the data set names, DD statements are not needed in the CICS startup job. This technique can be used to add additional files to an RFS pool or to add additional RFS filepools without restarting your CICS system.

The FILEPOOL DEFINE commands in member, CICSTART in data set CICSTS32.REXX.SCICEXEC have two purposes. The first is to define the names of the supplied filepools. They are: POOL1 and POOL2. You can modify these names to your installation standards. They can be from 1 to 8 characters. They should not contain special characters, “.” or “\”. The second purpose is to associate the filepool ids to the FCT definitions for the VSAM data set used for its directory and the first VSAM file used for data storage.

If you want to add additional RFS filepools to your system you need to add RDO definitions and add FILEPOOL DEFINE commands to your CICSTART member. If you intend to allow users to add RFS files to the new filepool you must define the filepool to include a \USERS directory.

To make these new filepools available for use, you must restart CICS. However, you can also add filepools while your CICS system is active as follows.

- add the RDS definitions for the new files and define them using a batch job
- then get an authorized user to enter the FILEPOOL DEFINE command and the FILEPOOL FORMAT command
- be sure to modify CICSTART or your new definitions are lost when you restart your CICS system

## TD queues needed for IMPORT and EXPORT commands

The REXX Development System uses dynamic allocation to IMPORT members from a partitioned data set or to EXPORT RFS files to a partitioned data set. The member CICRDOD in data set CICSTS32.REXX.SCICJCL defines three Transient Data entries used as input for IMPORT and 3 Transient Data entries for output for EXPORT. This allows three users to concurrently IMPORT and three users to concurrently EXPORT from and to partitioned data sets.

Modify the number of TDQ entries to suit your needs, but you should allow for at least one input and one output entry. The TDQUEUE NAME must begin with REX and be suffixed with a valid character. Ensure there are no other applications using TDQUEUE names that begin with REX because IMPORT/EXPORT will use them and may cause files to become corrupted.

## SQL definitions used for authorizing transactions to use DB2

The member CICRDOD in data set CICSTS32.REXX.SCICJCL authorizes the transactions, REXX, EDIT, FLST, and DXB0 to use the DB2 plan. The first three of these transactions are REXX for CICS transactions and the DXB0 transaction is added if you have OfficeVision/MVS and want to use DB2 interface calls which may run under the OV/MVS transaction id. If you choose to modify the supplied transactions for the REXX Development System, you need to modify the DB2 entry definitions also.

If you implement new transactions which use the DB2 interface code then you should also add these DB2 entry definitions to your RDO group.

## Special exec data sets used by the REXX Development System

There are three data set concatenations which are used by the REXX Development System which have no FCT entries. They are the DD names, CICCMDS, CICEXEC, and CICUSER. These data sets are partitioned data sets and are accessed using MVS facilities.

### CICCMDS

The CICCMDS DD name concatenation should start by referencing the data set CICSTS32.REXX.SCICCMDS. This data set contains those execs which implement REXX Development System authorized commands. Only authorized users or execs authorized to use authorized commands may access these execs. If you choose to extend the REXX Development System with your own authorized commands, you should concatenate your data set to this DD name concatenation.

### CICEXEC

The CICEXEC DD name concatenation should start by referencing the data set CICSTS32.REXX.SCICEXEC. This data set contains those execs which are supplied by the REXX Development System that use authorized commands. If you choose to extend the REXX Development System with your own execs which use authorized commands then you should concatenate your data set to this DD name concatenation.

### CICUSER

The CICUSER DD name concatenation should start by referencing the data set CICSTS32.REXX.SCICUSER. This data set contains those execs which are supplied by the REXX Development System that do not use authorized commands. If you choose to extend the REXX Development System with your own execs which do not use authorized commands then you should concatenate your data set to this DD name concatenation.

The facilities used to access these data set concatenations use CICS WAIT EXTERNAL capabilities to avoid placing the CICS region into a wait.

## Special USER ids and their usage

It is recommended that external security be used in the CICS environment. This is needed because individual user's information is maintained by the REXX Development System by the user's userid designation. Each user should have his own identification and there should not be two users signed on to the REXX Development System with the same userid at the same time. Two users with the same userid operating at the same time could have unusual results.

If a user is not signed on to the CICS region then the special userid of “\*RCUSER\*” is used to access the RLS and RFS facilities.

Authorized users are identified to the REXX Development System through the AUTHUSER command. This command is an authorized command and can only be used by an authorized user or an exec which is authorized to use authorized commands. CICSTART is such an exec because it resides in the CICEXEC DD name concatenation.

The member, CICSTART in the supplied data set CICSTS32.REXX.SCICEXEC then should be modified to contain an AUTHUSER statement to identify at least one userid which should be an authorized user. It is important to add the AUTHUSER statement after the existing AUTHUSER statement for RCUSER. For example:



```
'AUTHUSER RCUSER'  
  IF RC ^= 0 THEN EXIT RC  
'AUTHUSER your-userid'  
  IF RC ^= 0 THEN EXIT RC
```

You could also call another exec of your choosing, within the CICEXEC concatenation, which could contain the userids of those users who should be authorized users.

### Other considerations

The member, CICSTART in data set CICSTS32.REXX.SCICEXEC, contains default definitions for the REXX Development System. It is executed when the first user who executes a transaction that uses the CCREXD program after the CICS system has been started. This exec should be updated with any changes in customization if those changes are required to be effective across CICS executions.

The REXX Development System provides the capability to execute execs in either pseudo-conversational or conversational mode. The system default for conversational mode is specified with the SETSYS PSEUDO statement in the member, CICSTART in the data set CICSTS32.REXX.SCICEXEC. The default supplied provides for pseudo-conversational to be used. CICSTART must run in conversational mode because the system has not yet been initialized enough to ensure correct operation.

The CICSTART member also contains EXECLOAD commands which are commented as shipped. EXECLOADing execs will reduce the amount of storage used by the REXX Development System because users will then share the same exec. Also performance may be increased because these execs will not have to be loaded into CICS memory each time they are executed. EXECLOADed execs are used before any other execs. Therefore, if you EXECLOAD an exec "TEST.EXEC" and you have an exec in your RFS current directory by the same name, you will not be able to execute your RFS copy. Care should then be used both in the naming or your execs and EXECLOADing them. The authorization associated with the special DD names is maintained when execs are EXECLOADed from those DD name concatenation.

### Online HELP facility

An online HELP facility is provided which may be used as an example of the REXX/CICS panel facility. It provides the means to search and display the LIST3270 manual which is supplied with the product. There are several steps which must be done in order to activate the online help.

**Note:** If PTF maintenance has been applied that affects data sets for the procedure outlined below, then you should use the target library; otherwise, the distribution library should be used.

First, modify member CICSTART in the data set CICSTS32.REXX.SCICEXEC to reflect the correct RFS filepool and path where the online help files should be. If you choose to use the default no changes to CICSTART are necessary.

Next copy the supplied data set CICSTS32.REXX.SCICDOC to a data set whose highest level data set qualifier matches the userid of the user who will be executing the CICHPREP exec. The reason for this is the supplied security exit for the REXX/CICS IMPORT and EXPORT commands checks the highest level qualifier and it must match the userid for the user who issues the IMPORT command.

Next copy the supplied data set CICSTS32.REXX.SCICPNL to a data set whose highest level data set qualifier matches the userid of the user who will be executing the CICHPREP exec. This data set contains the panel definitions which are used by the online help. They must be IMPORTed into the RFS filepool and path defined for the online help.

Next sign onto REXX/CICS using the REXX transaction id. If you have changed the default transactions, this is the transaction id associated with the CICRXTRY exec. Issue the command, 'EXEC CICHPREP'. Follow the instructions issued by the exec. This exec will read the LIST3270 format of the manual from the data set you name, into the RFS directory specified in CICSTART. It also splits the manual into multiple files for usage by the online help. Additionally, the panels used by the online help are IMPORTed into the RFS system.

The online HELP facility is now ready for use.

There are several ways the user may access the online HELP.

Enter 'HELP' on the command line from the interactive environment and a table of contents is displayed. You also may enter this command from the command line of the REXX/CICS editor or the REXX/CICS filelist facilities.

Entering 'HELP xxxxx' searches the INDEX of the manual for the xxxxx entry. If found you are taken directly to that section of the manual.

There also is a HELP key defined for the editor and the filelist facility. It is defined in the customizing macros for the editor and the filelist facilities. The supplied default for this key is PF1. You may choose to modify the supplied default by modifying these profiles.

---

## Setting up a CICSplex SM address space (CMAS)

This section describes the steps you must perform in order to make a CICSplex SM address space (CMAS) operational. These steps consist of:

- “Before you begin”
- “CICSplex SM auxiliary storage usage” on page 253
- “Preparing to transmit generic alerts to NetView” on page 254
- “Preparing to start a CMAS” on page 254
- “CMAS-related CICS SIT parameters” on page 263
- “Shutting down a CMAS” on page 273
- “Restarting a CMAS” on page 274.

For a summary of the CMAS setup tasks that you can refer to while performing them, see “CICSplex SM setup checklist and worksheets” on page 9.

### Before you begin

Before you begin, check the IEASYSxx member of SYS1.PARMLIB that you use for MVS initialization and make note of the initialization values that are referenced during installation. For details about these values, see “Noting IEASYSxx values for CICSplex SM” on page 100.

If you are converting your CICSplex SM system or systems from a previous release to CICSplex SM for CICS TS for z/OS, Version 3.2, you should read the *CICS Transaction Server for z/OS Migration from CICS TS Version 3.1* manual.

In CICS Transaction Server for z/OS, Version 3 Release 2 a CICSplex SM CMAS will run only in a CICS system at the same release level. For example, a CICS TS 3.2 CMAS runs only in a CICS TS 3.2 region. During startup the CMAS checks the CICS release level and terminates with message EYUXL0142 if the releases do not match.

**Note:** These changes have no effect on the managed CICS systems.

For details on applying corrective or preventive maintenance to CICSplex SM, see Chapter 29, “Applying service to CICS Transaction Server for z/OS,” on page 207.

Take note of the information in the *CICSplex System Manager Concepts and Planning* manual about appropriate uses of a CMAS.

## CICSplex SM auxiliary storage usage

When a CMAS is initialized, up to 9 MVS dataspace are created. These dataspace are used by CICSplex SM to allow quick access to data from a CMAS and the MASs attached to it. Although the dataspace are logically owned by the CMAS, they are physically owned by the ESSS address space (EYUX320). The dataspace are deleted when the CMAS (that logically owns the dataspace) and all local MASs that are attached to that CMAS are terminated. The dataspace are recreated when the CMAS is initialized again.

The size of the dataspace is dependent upon the amount of work (end-user interface, workload management, MAS resource monitoring, and real-time analysis processing) the CMAS is performing and the number of MASs connected to the CMAS. The size may range from 20MB of storage in a relatively idle CICSplex SM configuration to well over 100MB of storage in a configuration that is complex in both the number of MASs and the amount of work requested. If you do not prepare for such an increase in storage usage, you may encounter auxiliary storage shortages when you first start to use CICSplex SM.

As an effort to prevent such auxiliary storage shortages, you should ensure that your auxiliary storage capabilities can handle an increase of 100MB of storage within the environment. Additionally, you can monitor CICSplex SM's dataspace usage by using an external monitor package to determine the amount of storage the EYUX320 job uses.

**Note:** If you contact IBM support personnel because of auxiliary storage shortages, they may ask you to use the CICSplex SM online debugging transactions (COD0 and CODB) to evaluate the storage use of EYUX320. For information about the COD0 and CODB transactions, refer to the *CICSplex System Manager Problem Determination* manual.

If auxiliary storage shortages do occur, you can alleviate the problem by either dynamically increasing your auxiliary storage capability or by causing CICSplex SM to free the allocated dataspace, as follows:

- To dynamically increase auxiliary storage capacity, allocate an additional page data set, then use the MVS console command PAGEADD to make the new page data set available.
- To cause CICSplex SM to free the allocated dataspace, first terminate the CICSplex SM agent in all local MASs connected to the CMAS. To do this, use the COSH transaction for each MAS, or if the MAS is a WUI server use the COVC and COSH transactions.

If a local MAS is acting as a CICSplex SM WLM TOR, and the DTR program is specified as EYU9XLOP for that MAS, you must change the DTR program from EYU9XLOP before you can use the COSH transaction against that MAS. (For example, you can change it to the IBM default program DFHDYP.)

After the CICSplex SM agent is terminated in all local MASs, terminate the CMAS itself using the COSD transaction.

After the auxiliary storage capability is increased, you can restart the CMAS. To reconnect any local MASs that remained active after the CICSplex SM agent was stopped, execute the COLM transaction within those CICS regions.

You can execute COLM using a modify command from the CONSOLE.

## Preparing to transmit generic alerts to NetView

You can have the real-time analysis (RTA) component of CICSplex SM transmit generic alerts to an IBM NetView system when one or more user-defined conditions occur during analysis.

To be sure that a NetView system is ready to receive the alerts, use the NPDA command

```
DFILTER AREC
```

to verify that the Event Type record IMPD is being passed to the NetView database in the NetView system.

The resulting list should show an ACTION of PASS for ETYPES of IMPD, and RSLV.

If it is necessary to add these record types to the filter, you can issue the following NPDA commands:

```
SRFILTER AREC PASS E IMPD  
SRFILTER AREC PASS E RSLV
```

If the name of the NetView Alert Receiver has been changed from the default value (NETVALRT), the CMAS only CICSplex SM system parameter ALERTRCVR may be used to specify the required name. See “CICSplex SM system parameters” on page 305 for details of the ALERTRCVR parameter.

## Preparing to start a CMAS

There are several ways to start a CMAS.

You can start a CMAS:

- When an MVS system is IPLed.

To use this method:

- Verify that the CMAS startup procedure is in a system procedure library, such as SYS1.PROCLIB.
- Verify that the CMAS startup procedure is in the 'Started Tasks' table of the external security manager (ESM).
- Change the COMMNDaa member that is referenced by the IEASYSxx member of SYS1.PARMLIB (as in “Noting IEASYSxx values for CICSplex SM” on page 100) to include a START command for the CMAS.

The START command to be included is described in “START command for a CMAS” on page 269.

- From the system console.

To start a CMAS from the system console:

- Verify that the CMAS startup procedure is in a system procedure library, such as SYS1.PROCLIB.
  - Verify that the CMAS startup procedure is in the 'Started Tasks' table of the external security manager (ESM).
  - Have the operator issue the START command described on page “START command for a CMAS” on page 269.
- As a batch job.

To start a CMAS as a batch job:

- Verify that the CMAS startup procedure is in a system procedure library, such as SYS1.PROCLIB.
- Construct a job stream to invoke the CMAS procedure.
- Submit the job to invoke a CMAS.

No matter which method you use to start a CMAS, be sure to verify that the procedure references the appropriate:

- CICS SIT parameters, as described on page “CMAS-related CICS SIT parameters” on page 263.
- CICSplex SM startup parameters, as described on page “CICSplex SM system parameters” on page 305.

**Note:**

1. Because the job of the CMAS is to manage a MAS, it is important that the CMAS have the ability to process data with a higher priority than the MAS. Therefore, when the MVS image is running in workload management goal mode, the CMAS jobs should be defined to the MVS service class, SYSSTC, for optimal performance. Failure to do so may result in severe performance problems for CICSplex SM.
2. After starting a CMAS for the first time, you must configure the CMAS to your environment. This includes establishing the CICSplexes it is to manage and any communication links that are needed between this CMAS and another CMAS. For additional information about this, see the *CICSplex System Manager Administration* manual.

A sample procedure that you can use to start a CMAS is supplied in the member EYUCMASP. This member was generated when you ran the DFHISTAR job., The member is stored in the library you specified on the LIB parameter of the DFHISTAR job.

You must create the data sets for this CICS region. JCL to create the CICS region data sets for the CMAS is supplied in member EYUCMSDS of CICSTS32.CPSM.XDFHINST. This member was generated when you ran the DFHISTAR job.

Figure 22 on page 256 illustrates segments of the EYUCMASP procedure that highlight the additional CICSplex SM requirements.

```

| //EYUCMASP PROC DSNCSO=CICSTS32.CPSM.CMAS01.DFHCSO, CSD Data Set name
| //          DSNTBL=CICSTS32.CPSM.RGNLOAD, CICS Table Module library
| //          RGNHLQ=CICSTS32.CPSM.CMAS01,          CICS Region DSN qualifier
| //          CICSJHLQ=CICSTS32.CICS
| //          CPSMHLQ=CICSTS32.CPSM
| //          PRMLIB=CICSTS32.XDFHINST
| //          CICSJPRM=EYUCMSO,          CPSM Parameters
| //          CPSMJPRM=EYUCMSO
| //CICS EXEC PGM=EYU9XECS,          CMAS Startup program
| //          PARM='SYSIN',          CICS Parameters location
| //          REGION=OK          Region Size
| // *
| //STEPLIB DD DISP=SHR,DSN=&CPSMHLQ..SEYUAUTH
| //          DD DISP=SHR,DSN=&CICSJHLQ..SDFHAUTH
| //DFHRPL DD DISP=SHR,DSN=&CPSMHLQ..SEYULOAD
| //          DD DISP=SHR,DSN=&CICSJHLQ..SDFHLOAD
| //SYSIN DD DISP=SHR,DSN=&PRMLIB,(&CICSJPRM)
|
| :
| //EYULOG DD SYSOUT=*
|
| :
| //EYUDREP DD DISP=SHR,DSN=CICSTS32.CPSM.EYUDREP.cmasname
| //EYUPARM DD DISP=SHR,DSN=&PRMLIB(&CPSMJPRM)

```

Figure 22. CMAS-specific JCL requirements

Review the following statements in the sample JCL that are illustrated in Figure 22. Verify that the JCL has been modified so that the:

**EXEC PGM=EYU9XECS statement**

Starts the CMAS and either verifies the existence of, or creates, the ESSS. EYU9XECS, the CMAS startup program, must be run in order for a CMAS to initialize.

**STEPLIB DD statement**

Includes the CICSTS32.CPSM.SEYUAUTH authorized load library.

**DFHRPL DD statement**

Includes the CICSTS32.CPSM.SEYULOAD load library. Include the load library that contains the CICS resource definition table load modules. These need to be link edited into a user-supplied load library, which needs to be specified in the DFHRPL concatenation.

You should not include application load libraries in the DFHRPL concatenation.

**SYSIN DD statement**

Identifies the library member containing the CICS system initialization table override parameters.

**EYULOG DD statement**

Identifies the log to which messages from the CMAS and its associated managed application systems (MASs) are to be directed.

When you are using a sequential data set for the EYULOG, allocate 3 primary cylinders and 1 secondary cylinder.

**EYUDREP DD statement**

Identifies the library to be used as the data repository by the CMAS, where:

**cmasname**

Is the name you specified for the CMASNAME parameter on the

DFHISTAR job. The CMASNAME value is used by EYU9XDUT in order to create the CICSPlex SM data repository.

**EYUPARM DD statement**

Identifies the library that contains the CICSPlex SM system parameters.

**Creating and managing the CICSPlex SM data repository**

This section describes the following:

- “Creating the CICSPlex SM data repository”
- “Populating the CICSPlex SM data repository” on page 259
- “Converting the CICSPlex SM data repository” on page 260
- “Expanding the CICSPlex SM data repository” on page 261
- “Taking backups of the CICSPlex SM data repository” on page 261

You can use the post-installation job EYUCMSDS to create the CICSPlex SM data sets. The EYUCMSDS job is generated when you run the DFHISTAR job, as described in “CICSPlex SM post-installation members” on page 217. The job is stored in the library you specified on the LIB parameter of the DFHISTAR job.

**Creating the CICSPlex SM data repository:** The CMAS-related data set is the data repository. Each CMAS must have a unique data repository associated with it. The data repository contains the CICSPlex SM administration definitions applicable to its associated CMAS.

**Note:** The data repository is a critical component of CICSPlex SM system management. It is imperative that you take regular backups that are associated with each CMAS in your environment.

It is defined to CICS as being a recoverable file which participates in SYNCPOINT and SYNCPOINT ROLLBACK operations. The CMAS must have a CICS system log in order for these operations to operate correctly. Do not, therefore, run a CMAS with a system log that is defined as type DUMMY as this would compromise data integrity on the CICSPlex SM data repository.

To create the data set that contains the data repository, you can use the post-installation job EYUCMSDS.

If you will be running multiple CMASs in the same MVS image, you must create a data repository for each CMAS. You can edit and resubmit the DFHISTAR job, which generates the EYUCMSDS post-installation job.

To ensure you do not overwrite your existing customized jobs, you can use the SELECT parameter, as described in “Setting up a CICSPlex SM address space (CMAS)” on page 252. Alternatively, if you prefer not to resubmit DFHISTAR, you can edit the EYUCMSDS job, giving the SYSIDNT and CMASNAME parameters unique names each time you run the job.

The EYUCMSDS job includes the following steps relating to the creation of the data repository:

**DELDREP**

This step deletes the data repository set. It allows you to resubmit the job.

**DEFDREP**

This step allocates the VSAM KSDS cluster for the data repository data set:

dsindex.EYUDREP.cmasname

where:

**dsindex**

Is defined by the DSINFO parameter of the DFHISTAR job.

**cmasname**

Is defined by the CMASNAME parameter of the DFHISTAR job.

**Note:** CICSplex SM does not support VSAM records that span control intervals. Make sure that the IDCAMS job that you use to create a CICSplex SM data repository does not specify the SPANNED parameter. You should accept the IDCAMS default of nonspanned records.

**DREPINIT**

This step is used to set up the data repository for a CICS TS for z/OS, Version 3.2 CMAS.

Step DREPINIT is generated in job EYUCMSDS if you did not specify a value with the OLDDREP parameter when you ran the DFHISTAR job. This step executes EYU9XDUT to initialize the new data repository that was allocated by step DREPALLOC. The new data repository does not contain any records from a previous version of CICSplex SM. The EYU9XDUT utility uses the following parameters for step DREPINIT:

**CMASNAME=xxxxxxx**

- You cannot change this name after the data repository is initialized.
- This name must be unique within the CICSplex SM environment. It should not be the same as the name of another CMAS, a CICSplex, a CICS system, or a CICS system group.
- Position 1 must be alphabetic or national, and cannot be numeric.
- Positions 2 through 8 can be alphabetic, national, or numeric.

**SYSID=xxxx**

- You cannot change this identifier after the data repository is initialized.
- This value must match the SYSIDNT (SIT parameter) for the CMAS; see “CMAS-related CICS SIT parameters” on page 263.
- This value must not be the same as the SYSID for any other CMAS or CICS system that is defined to CICSplex SM.
- Positions 1 through 4 can be alphabetic, national, or numeric.

**TIMEZONE=x**

where x must be a single alphabetic character (B through Z), representing one of the Greenwich time zone codes. See the *CICSplex System Manager Administration* for more information on attribute definitions.

**ZONEOFFSET=nn**

where nn must be a two-digit numeric value (00 through 59), representing an adjustment (offset) to the TIMEZONE.

**DAYLIGHT=x**

where x must be a single character (Y or N), representing daylight saving time.



For information about defining the TIMEZONE, ZONEOFFSET, and DAYLIGHT parameters, see the *CICSplex System Manager Administration*.

The DREPINIT step requires a DD statement for EYUXDPRM which sets up the WUI parameters as follows:

**WUI=YES|NO**

where the default of WUI=YES creates the CICSplex SM resource definitions required to start a WUI server and its CICSplex. This allows you to create the resource definitions required to start a WUI server and its CICSplex when you create the data repository. If you need to add a WUI to an existing system, use the EYU9XDBT or BATCHREP utilities. See the *CICSplex System Manager Administration* for more information about these utilities.

**WUIAPPLID=xxxxxxx**

where xxxxxxx must be alphabetic, national or numeric characters, specifying the APPLID allocated to a WUI. The first character must be alphabetic or national. This parameter is mandatory if WUI=YES is specified.

**WUINAME=xxxxxxx**

where xxxxxxx must be alphabetic, national or numeric characters, specifying the name allocated to a WUI. The first character must be alphabetic or national. If this is not specified, WUINAME takes the value specified for WUIAPPLID.

**WUIPLEX=xxxxxxx**

where xxxxxxx must be alphabetic, national or numeric characters, specifying the name allocated to a WUI CICSplex. The first character must be alphabetic or national. The default is created from the characters WUIP, followed by the CMSSYSID. For example, using the default CMSSYSID, CM01, the default WUIPLEX name would be WUIPCM01.

**WUISYSID=xxxx**

where xxxx must be alphabetic, national or numeric characters, specifying the name allocated to a WUI system identifier. The first character must be alphabetic or national. This parameter is mandatory if WUI=YES is specified.

**DREPCNVT**

This step is generated if you specified the name of an existing data repository on the OLDDREP parameter. This step executes EYU9XDUT to convert existing data repository records from a previous release of CICSplex SM for use by CICSplex SM for CICS TS for z/OS, Version 3.2. All the records from the input data repository specified on the OLDDREP parameter are added to the new data repository that was allocated by step DREPALLOC. The input data repository is not modified. The EYU9XDUT utility uses the following parameter for step DREPCNVT:

**TARGETVER=0320**

where 0320 represents the version of the new output data repository.

See “Converting the CICSplex SM data repository” on page 260

**Populating the CICSplex SM data repository:** You can use the CICSplex SM-supplied extract routine EYU9BCSD to generate CICSplex SM resource definition records for each CSD record identified in your input file.

The output from EYU9BCSD is used to populate the data repository.

For more information about EYU9BCSD see the *CICSplex System Manager Managing Business Applications* manual.

**Converting the CICSplex SM data repository:** You can run the EYU9XDUT utility to convert the CICSplex SM data repository from any release of CICSplex SM to this release, and back again. That is, you can upgrade the data repository to the current release of CICSplex SM, and you can convert it back to a previous release. For example, after you have upgraded to CICSplex SM for CICS Transaction Server for z/OS, Version 3 Release 2, you can convert the data repository back to CICSplex SM for CICS Transaction Server for z/OS Release 2.3 for use with a CMAS running that level of CICSplex SM code.

The conversion is controlled by the TARGETVER parameter and the DD statements you use to execute EYU9XDUT. To convert between a previous release and CICSplex SM for CICS TS for z/OS, Version 3.2 requires the EYUDREP, NEWREP, and BYPASS DD statements in the JCL. The EYUDREP statement must reference an existing input data repository, and the NEWREP statement must reference the output data repository.

CRESxxxx is a CICSplex SM Topology Manager object that describes instances of a resource within a CICS system. All CRESxxxx resource table objects are deleted during the conversion process. These deleted records are rebuilt the first time the MAS connects or joins to the CMAS. The following message is then output to the SYSPRINT DD statement with a count of how many records were not converted:

```
EYUXD0708I CRESxxxx Resource Table nnnnn records not converted
```

EYU9XDUT Return codes:

**0** execution completed normally.

This return code is accompanied by message: EYUXD0702I Repository successfully converted indicating the repository was successfully converted. This message may also be preceded by the message: EYUXD0708I CRESxxxx Resource Table nnnnn records not converted

**4** some records could not be converted.

This return code is accompanied by message: EYUXD0706W DDNAME NEWREP nnnnn records were bypassed indicating the repository was successfully converted but some records were bypassed and not written to the new repository. These records are written to the BYPASS DD statement. This message can also be preceded by message: EYUXD0708I CRESxxxx Resource Table nnnnn records not converted

**8** EYU9XDUT has failed.

Check the joblog for error messages that indicate the cause of the error. For example, DD statement missing, or invalid NEWREP dataset.

**Note:** Shutdown the CMAS that is using EYUDREP before running EYU9XDUT, otherwise, X'A8' is returned when the dataset is opened.

Use JCL similar to that in Figure 23 on page 261 to convert from Release 2 of CICSplex SM to CICSplex SM for CICS TS for z/OS, Version 3.2. This sample JCL assumes that you have already allocated the VSAM cluster that is referenced by the NEWREP DD statement. The file identified by the NEWREP DD statement

```

I          must be an empty file.

I //DREPCNVT EXEC  PGM=EYU9XDUT,
I //          PARM=(' TARGETVER=0320' )
I //STEPLIB  DD   DISP=SHR,DSN=CICSTS32.CPSM.SEYUAUTH
I //EYUDREP  DD   DISP=SHR,DSN=CPSM320.EYUDREP.cmasname
I //NEWREP   DD   DISP=OLD,DSN=CICSTS32.CPSM.EYUDREP.cmasname
I //BYPASS   DD   SYSOUT=*
I //SYSPRINT DD   SYSOUT=*

```

Figure 23. Example JCL to run EYU9XDUT

In the JCL, use a value for TARGETVER that indicates the release you are converting to. For example:

To convert to..	Use:
Release 3	TARGETVER=0130
CICSplex SM for CICS TS Release 3	TARGETVER=0140
CICSplex SM for CICS Transaction Server for z/OS Release 2.1	TARGETVER=0210
CICSplex SM for CICS Transaction Server for z/OS Release 2.2	TARGETVER=0220
CICSplex SM for CICS Transaction Server for z/OS Release 2.3	TARGETVER=0230
CICSplex SM for CICS Transaction Server for z/OS Release 3.1	TARGETVER=0310
CICSplex SM for CICS Transaction Server for z/OS Release 3.2	TARGETVER=0320

When the data repository is updated from a previous release to the current release, information stored in the earlier version is carried over to the current version of the data repository. When the data repository reverts back from the current release to a previous release, and the record in which information is changed is identical in both releases, the changes are carried back to the earlier version. Likewise, if a field in a record exists in both releases, later changes to the information in a field are carried back to the earlier version. However, when the data repository is converted from the current release to a previous release and either the record or a field in it did not exist in the previous release, the information contained in the later version is permanently lost.

(For more information about the data repository, see “Creating the CICSplex SM data repository” on page 257 and “Expanding the CICSplex SM data repository.”)

**Expanding the CICSplex SM data repository:** The CICSplex SM data repository may fill up and require expansion. To expand the CICSplex SM data repository, use the IDCAMS utility REPRO function. An example of the JCL to do this is in member EYUJXDRP of the CICSTS32.CPSM.SEYUSAMP library.

In that JCL, on the RECORDS(xx,yy) statement. You should specify a primary (xx) and a secondary (yy) value that are appropriate for your environment. The initial values are 500 and 3000.

**Taking backups of the CICSplex SM data repository:** The CICSplex SM data repository is defined to CICS as a VSAM file called EYUDREP. As the data set is accessed via CICS File Control, all the normal CICS methods of taking backups of VSAM data sets for disaster recovery purposes are available for use with the data repository.

You may use the following techniques for taking copies of the data repository, and for restoring the data repository after a data set failure.

- Use HSM, or DSS, or other utilities to take copies while the associated CMAS is not running, possibly using the Concurrent Copy technique to reduce the time during which the repository must be unavailable.
- Use HSM or DSS to take copies while the associated CMAS is running using the Backup While Open technique, and possibly also using the Concurrent Copy technique, which improves the ease of use of Backup While Open. This requires a forward recovery log (see “Defining a forward recovery log for the data repository”).
- Use HSM or DSS to restore the data set after a data set failure.
- Use a Forward Recovery product, such as CICS VSAM Recovery (CICS/VR), to reapply updates that were made to the data set after the most recent copy was taken. This requires a forward recovery log.
- Use remote site recovery techniques if you need an up-to-date copy of the data set at a remote site for disaster recovery purposes. This requires a forward recovery log.

The *CICS Recovery and Restart Guide* provides information on all the terms referred to above. In particular, it provides information about forward recovery logs, forward recovery, the CICS/VR product, Backup While Open, Concurrent Copy and its associated hardware prerequisites, taking back ups of data sets, restoring data sets from backup copies, and remote site recovery.

*Defining a forward recovery log for the data repository:* The data repository is defined in the CMAS as a VSAM file called EYUDREP. CICSplex SM provides a default definition that defines this file as not having an associated forward recovery log, and therefore as not being eligible for forward recovery.

If you use forward recovery, you require a journal logstream. Defining and setting up CICS log streams is described in the *CICS System Definition Guide*.

If you wish to use Forward Recovery, Backup While Open, or Remote Site Recovery, you will need to change the definition of EYUDREP. You will need to specify the following keywords on the definition of EYUDREP to define it as having a forward recovery log:

```
RECOVERY(ALL)
FWDRECOVLOG(nn)
```

where *nn* is a number between 1 and 99.

See “Overriding the dynamically created CICS resource definitions for CICSplex SM” on page 227 for an example of how to do this.

The default definition of EYUDREP also does not define the repository as being eligible for Backup While Open. To make the repository eligible for Backup While Open, you should specify the following keywords:

```
RECOVERY(ALL)
FWDRECOVLOG(nn)
BACKUPTYPE(DYNAMIC)
```

where *nn* is a number between 1 and 99.

The RECOVERY, FWDRECOVLOG, and BACKUPTYPE parameters of DEFINE FILE are described fully in the *CICS Resource Definition Guide*.

**Note:**

1. You should not change any keywords on the EYUDREP definition other than RECOVERY, FWDRECOVLOG and BACKUPTYPE. In addition, you must never set RECOVERY(NONE). Setting RECOVERY(NONE) would cause repository corruption after transaction or CMAS failures.
2. You should not change the recovery options of the EYUDREPN FILE definition. This definition is used when CPSM determines that Data Repository file operations do not require logging. It is normal to receive LSR pool messages for EYUDREPN during CMAS initialization and you should ignore them. There should be no DD statement in the CICS JCL for EYUDREPN and EYUDREPN should not be associated with a data set name.
3. If CPSM Data Repository initialization fails (as reported by message EUIXD0105E) and the cause is due to the EYUDREP data set requiring Batch Backout (for example CICS issues message DFHFC0921), you must recover the EYUDREP data set and then delete and redefine the CMAS Local and Global catalogs in order to reset the CICS backout required status for the data set.
4. Requesting Backup While Open for the CICSplex SM data repository data set via the IDCAMS DEFINE CLUSTER definition within the ICF catalog is not supported.

**Editing CICSplex SM system parameters**

Member EYUCMS0P, in the CICSTS32.CPSM.XDFHINST (or TDFHINST) data set, contains sample parameters for a CMAS; this member must be edited. (See “CICSplex SM system parameters” on page 305 for a detailed description of each parameter.)

When the CMAS is to connect to a MAS for which security is active (the CICS SIT parameter for the MAS is SEC=YES), the CMAS must have CICSplex SM security active. When CICSplex SM security is not activated in the CMAS, the connection between the CMAS and the MAS cannot be established. If this is attempted, the following message is issued to the console, the CMAS joblog, and the CMAS EYULOG:

```
EYUCR0007E Security mismatch between CMAS cmasname and
           MAS masname. Connection terminating.
```

To activate CICSplex SM security in the CMAS, you must specify the CICSplex SM system parameter SEC=YES). The default is SEC(NO). (For more information about the SEC parameter, see “CICSplex SM system parameters” on page 305.) Specifying SEC=YES in the CICS SIT parameters for the CMAS does not affect CICSplex SM security.

**CMAS-related CICS SIT parameters**

Member EYUCMSSP is supplied, uncustomized, in TDFHINST and customized by DFHISTAR, in XDFHINST. This contains the CICS system initialization table (SIT) parameters for a CMAS.

Table 17 on page 264 identifies the CMAS-related CICS SIT parameters.

**Note:**

1. When the second column in the table contains an asterisk, before you start a CMAS you should supply your own value for the parameter listed in the first column.

2. When the second column of the table does not contain an asterisk, do not change the value of the parameter in the first column.

Table 17. CICS SIT parameters for a CMAS

Parameter	Your value	Explanation
AIEXIT=DFHZATDX		VTAM terminal autoinstall program.
APPLID=	*	VTAM application ID for this CICS, which is acting as a CMAS. Used as CMAS name when NAME(value) is not specified as a CICSplex SM system parameter.
AUXTR=ON		Auxiliary trace - Exception records.
AUTORESETTIME=YES		Time-of-day synchronization.
AUXTRSW=ALL		Continuous auxiliary trace switching.
CICSSVC=216	*	CICS SVC installed in LPA.
CPSMCONN=CMAS		Initialize this region as a CMAS
CSDACC=READWRITE		Enable read and write updates to CSD.
CSDRECOV=ALL		CSD forward recovery and backout.
DFLTUSER=CICSUSER	*	Non-CESN RACF user Id.
DSALIM=5M		Limit of DSA storage below 16MB. This is a minimum initial value.
DUMPDS=A	*	Transaction dump data set.
DUMPSW=NEXT	*	Switch to next transaction dump data set.
EDSALIM=100M		Limit of EDSA storage above 16MB. This is a minimum initial value. See "Controlling CICS storage in a CMAS" on page 269 for additional information.
FCT=NO		No File control table.
GMTEXT='CICSplex System Manager - CICS Transaction Server for z/OS'	*	Default logon message.
GRPLIST=DFHLIST		CICS group list. See "Overriding the dynamically created CICS resource definitions for CICSplex SM" on page 227 for additional information.
ICV=100		Region exit interval.
ICVR=20000		Runaway task interval. <b>Note:</b> For a CMAS running on a small processor and having a large number of resources defined through BAS, this value may need to be increased to about 90000.
ICVTSD=1		Terminal scan delay interval.
INTTR=ON		Activate main storage trace.
IRCSTRT=YES		IRC started at system initialization.
ISC=YES		Load programs required for interregion or intersystem communications during initialization.
MXT=300		Maximum tasks to exist. This is a minimum initial value. See "Controlling CICS storage in a CMAS" on page 269 for additional information.
RENTPGM=PROTECT		Specifies that CICS will allocate ERDSA from readonly key 0 protected storage.

Table 17. CICS SIT parameters for a CMAS (continued)

Parameter	Your value	Explanation
SEC= {YES NO}		<p>Indicates whether CICS external security checking is performed for this CMAS. The CICS security checking is independent of the CICSplex SM external security checking (which is controlled with the CICSplex SM SEC system parameter) as specified in the EYUPARM DD. For information about the SEC CICSplex SM system parameter for CMAS, see “CICSplex SM system parameters” on page 305.</p> <p>The CICS external security checking SIT option can be specified in the normal way, along with the other CICS security related options (for example, XTRAN and XCMD).</p> <p>CICS command or resource security is not appropriate in a CMAS. The XTRAN SIT option is typically used to control access to the various CICSplex SM transactions used in a CMAS. For more information, see the <i>CICS RACF Security Guide</i>.</p> <p><b>Note:</b> No CICSplex SM supplied transaction definitions have CMDSEC=YES or RESSEC=YES. If this definition is changed, or the CMDSEC=ALWAYS or RESSEC=ALWAYS SIT parameters are set in a CMAS, and a CICSplex SM transaction receives a NOTAUTH response, results are unpredictable.</p>
SIT=6\$		System initialization table suffix.
SPOOL=YES		System spooling interface. Required when you are going to use the CICSplex SM batched repository-update facility.
START=AUTO		<p>You can normally specify START=AUTO and let CICS initialization decide the type of start to perform.</p> <p>The first time you start a CMAS, ensure the CICS global and local catalog data sets are newly initialized. Use DFHRMUTL and DFHCCUTL respectively, with AUTOINIT on the SET_AUTO_START parameter of DFHRMUTL. This makes sure that the CMAS performs an initial start, which installs the necessary CICS resource definitions and establishes CMAS-to-CMAS connections.</p> <p>Subsequently, you can manipulate the type of start for a CMAS by resetting the global catalog data set, using DFHRMUTL to specify either AUTOINIT or AUTOCOLD. For more information, see “Restarting a CMAS” on page 274.</p>
SUBTSKS=1		Use additional concurrent mode TCB.
SYSIDNT=	*	<p>CICS System Id.</p> <p><b>Note:</b> The SYSIDNT value must match the EYU9XDUT SYSID parameter value used to initialize the data repository being referenced by the EYUDREP DD statement.</p>
SYSTR=OFF		No system activity trace.
TCT=NO		No TCT needed.
TST=NO		No temporary storage table required.

Table 17. CICS SIT parameters for a CMAS (continued)

Parameter	Your value	Explanation
USERTR=ON		Enable user trace facility.
WRKAREA=2048		Bytes for Common Work Area.
I XAPPC=NO		RACF checking of APPC sessions.
I XDCT=NO		RACF checking of DCT entries.
I XEJB=NO		RACF checking of security roles.
XHFS=NO		Security checking of Web client access to z/OS UNIX files. For a CMAS you must specify XHFS=NO.
I XJCT=NO		RACF checking of JCT entries.
I XPSB=NO		RACF checking of PSBs.
XRES=NO		RACF checking of CICS document templates.
XRF=NO		No XRF support. <b>Note:</b> The extended recovery facility (XRF) is not supported because of the way in which a CMAS uses MVS data spaces.
I XTRAN=NO		RACF checking of transaction-attach.
I XTST=NO		RACF checking of TST entries.

## Controlling tasks in a CMAS

Many operations within a CMAS are achieved by multiple asynchronous tasks. This is especially true of operations that are performed between CMASs in a CMAS network, such as data repository synchronisation, workload management state sharing and single system image. Any of these operations, and others, can result in a number of interdependent asynchronous tasks being established or used to execute the request. The number of tasks that can be used is based on other factors, such as the size of a CMAS network, how many MASs are being managed, how many CICSplexes are defined, how much API activity is performed, the scope of WUI/API/RTA requests and so on for all the major functionality offered by CICPlex SM.

Even though a CMAS can self regulate its tasking model and has tolerance of delayed requests and responses through timeout mechanisms, MXT is a concept that is not applicable to controlling an interdependent multitasked asynchronous tasking model. Set inappropriately, you may also experience WUI hanging for long durations if one or more of the asynchronous tasks required to execute the requested function are delayed waiting for an MXT slot. It is therefore strongly recommended that MXT should be set to avoid **any** delays in task attachment.

As task usage within a CMAS grows with the additional requirements that are placed upon it, such as (but not limited to), increased use of the API, more CMASs, more MASs, new function use, and so on, you should set the MXT value to a level that continues to avoid MXT delays.

It is recommended that you monitor any MXT value for its relationship against the task activity within each CMAS at regular intervals. If the CMAS is starting to experience MXT delays, you should adjust the MXT value to avoid these delays.

To monitor the tasking activity within an individual CMAS, it is recommended that you collect and study the statistics generated by the CICS system that underlies the CMAS that it hosts. CICS transaction manager global statistics contain information



on the effect the MXT value has on task attachment. For more information and guidance on using CICS statistics see the *CICS Performance Guide*.

## **CMAS data set creation and customization**

DFHISTAR allows you to create and customize your CMAS data sets, according to the parameters you set when you submit the DFHISTAR job.

## **DFHISTAR post-installation members for a CMAS**

When you run DFHISTAR, with a SCOPE of POST or ALL, it creates the following post-installation members for a CMAS in the XDFHINST library:

- EYUCMSDS – creates and initializes all the data sets for a CMAS. EYUCMSDS includes steps to delete the data sets so that you can rerun the job, if required. These deletions are expected to fail the first time you run the job. EYUCMSDS contains the following steps:
  - DELDREP and DEFDREP delete and define a new CMAS data repository.
  - DREPINIT is included if you do not specify the DFHISTAR OLDDREP parameter. It initializes the new CMAS data repository using the EYU9XDUT utility. EYU9XDUT creates records on the data repository to define the CMAS and, by default, a WUI CICSplex (a WUI CICSplex will not be created if you specify the DFHISTAR WUI=NO option).
  - DREPCNVT is included if you specify the name of an existing data repository using the DFHISTAR OLDDREP parameter. It copies all the records from the existing repository to the new data repository, upgrading them to a format suitable for the CICS TS 3.2 release.
  - JES3DELA and JES3DEFA are included if you specify the DFHISTAR JES=JES3 option. They delete and define the CICS local catalog, the global catalog and the local request queue.
  - DELREGDS deletes the CICS data sets.
  - DEFTRACE defines the CICS auxiliary trace data sets, DFHAUXT and DFHBUXT.
  - DEFHTML defines the CICS DFHHTML data set.
  - DEFDMPS defines the CICS transaction dump data sets, DFHDMPA and DFHDMPB.
  - DEFTSTD defines the CICS auxiliary temporary storage data set, DFHTEMP.
  - DEFINTD defines the CICS intrapartition transient data set, DFHINTRA.
  - DEFLCD defines the CICS local catalog, DFHLCD.
  - INITLCD uses the DFHCCUTL utility to initialize the CICS local catalog.
  - DEFGCD defines the CICS global catalog, DFHGCD.
  - INITGCD uses the DFHRMUTL utility to initialize the CICS global catalog.
  - DEF LRQ defines the CICS local request queue data set, DFHLRQ.
  - JES3DELA and JES3DELB are included if you specify the DFHISTAR JES=JES3 option. They delete and define the CICS DFHCSD data set.
  - DELCSD deletes the CICS DFHCSD data set.
  - DEFCSO defines the CICS DFHCSD data set.
  - INITCSD uses the DFHCSDUP utility to initialize the DFHCSD data set.
- EYUCMSSP – CICS SIT overrides for a CMAS.
- EYUCMSOP – EYUPARM parameters for a CMAS.
- EYUCMASP – starts a CMAS.
- EYUCMASJ – JCL to start a CMAS. It executes EYUCMASP.

If you use the default values for the CICSplex SM parameters, the EYUCMASP PROC statement is as follows:

```
EYUCMASP PROC DSNCSO='CICSTS32.CPSM.CMAS01.DFHCSO',
             RGNHLQ='CICSTS32.CPSM.CMAS01',
             CICSJHLQ='CICSTS32.CICS',
             CPSMHLQ='CICSTS32.CPSM',
             PRMLIB='CICSTS32.XDFHINST'
             CICSJPRM=EYUCMSO,           CICS Parameters
             CPSMJPRM=EYUCMSO           CPSM Parameters
```

## Customizing post-installation jobs using DFHISTAR

You can use DFHISTAR to generate copies of the CMAS post-installation jobs for a different CMAS. DFHISTAR has a SELECT parameter that allows you to specify a new name for a copy of a post-installation job. It has the format:

```
SELECT jobname newname
```

where:

**jobname**

is the name of the job you want to regenerate

**newname**

is the name for the new copy.

You can specify more than one SELECT parameter to select multiple jobs to be regenerated in a single run of the DFHISTAR job. Once you include a SELECT parameter in the DFHISTAR job, only those jobs specified by the SELECT are generated.

For a CMAS with the name CMAS02 and a CICS system identifier of CM02 you can change your DFHISTAR parameters to specify the following:

```
CMASNAME CMAS02
CMSSYSID CM02
WUI YES
WUIPLEX WUIPCM02
WUINAME WUINCM02
WUISYSID WU02
SELECT EYUCMSO CM02CMO          JCL to create the data sets for CMAS02
SELECT EYUCMSO CM02CMO          CICS SIT overrides for CMAS02
SELECT EYUCMSO CM02CMO          CICSplex SM EYUPARM parameters for CMAS02
```

**Note:** Using these parameters, CM02CMO includes a step to initialize the CMAS data repository with the definitions for a WUI called WUINCM02, within a CICSplex called WUIPCM02.

You can then start CMAS CMAS02, using the procedure EYUCMASP, by specifying:

```
START EYUCMASP, DSNCSO='CICSTS32.CPSM.CMAS02.DFHCSO',
             RGNHLQ='CICSTS32.CPSM.CMAS02', CICSJHLQ='CICSTS32.CICS',
             CPSMHLQ='CICSTS32.CPSM', PRMLIB='CICSTS32.XDFHINST'
             CICSJPRM=CM02CMO, CPSMJPRM=CM02CMO
```

If you are using EYUCMASJ to start the WUI, you should edit it to specify the following:

```
CICSJPRM=CM02CMO, CPSMJPRM=CM02CMO
```

## Controlling CICS storage in a CMAS

A CMAS is a special application dedicated to the task of managing and controlling MASs. Even though a CMAS does this with an atypical tasking model and with extensive use of MVS dataspace storage, it still has major uses of the storage provided by the CICS system that hosts the CMAS. With an interdependent multitasked asynchronous tasking model there is a reliance on shared storage to perform the communication between the tasks and the functions being performed. Also, according to the requirements that are placed upon a CMAS, there is a large reliance on shared storage to perform the buffering of requests and responses to be transmitted via CMAS to CMAS and CMAS to MAS links. The shared storage to support such operational characteristics is managed by the CICS system that hosts the CMAS.

As CICS storage usage within a CMAS grows with the additional requirements that are placed upon it, such as, increased use of the API, more CMASs, more MASs, increased CMAS to CMAS network traffic and new function use, you should set the EDSALIM value to a level that provides the CMAS with the amount of storage it needs to perform its operations unimpeded. CMAS to CMAS and CMAS to MAS network traffic, in particular, is critical to WUI response times because large amounts of data may be awaiting shipment on any of the links that a CMAS communicates on

It is recommended that you monitor any EDSALIM value for its effect on the storage usage within each CMAS at regular intervals. If the CMAS is experiencing short on storage (SOS) or storage fragmentation or trending towards such conditions, you should consider increasing the EDSALIM value to meet the storage requirements of the CMAS. Storage fragmentation below a largest free area of 64KB will begin to adversely effect throughput and response times.

To monitor the storage usage within an individual CMAS, it is recommended that you collect and study the statistics generated by the CMAS CICS system. CICS storage manager global statistics contain information on the overall usage of CICS storage by the CMAS that it hosts. For more information and guidance on using CICS Statistics see the *CICS Performance Guide*.

## START command for a CMAS

The syntax of the command you can use to start a CMAS is:

```
START procname [,DSNCSD=dsn][,RGNHLQ=idx][,CICSHLQ=idx] [,CPSMHLQ=idx] [,PRMLIB=lib]
[,CICSPRM=mem] [,CPSMPRM=mem]
```

where:

### **procname**

Is the 1- to 8-character name of the procedure. EYUCMASP is the name of the sample procedure. It is supplied, uncustomized, in the TDFHINST library and also customized by DFHISTAR, in the XDFHINST library.

### **DSNCSD=dsn**

Specifies the name of the data set that contains the CSD file for the CMAS.

### **RGNHLQ=idx**

Specifies the high-level qualifier that is used with the DFHxxxx data sets that are unique to this CMAS.

### **CICSHLQ=idx**

Specifies the high-level qualifier that is used with the SDFHAUTH and SDFHLOAD libraries.

	<b>CPSMHLQ=idx</b>
	Specifies the high-level qualifier that is used with the SEYUAUTH and SEYULOAD libraries.
	<b>PRMLIB=idx</b>
	Specifies the name of the library containing the members identified by CICSPRM and CPSMPRM.
	<b>CICSPRM=mem</b>
	Identifies the member that contains the CICS SIT parameters. The EYUCMSSP sample is supplied, uncustomized, in the TDFHINST library and also customized by DFHISTAR, in the XDFHINST library.
	<b>CPSMPRM=mem</b>
	Identifies the member that contains the CICSplex SM system parameters. The EYUCMS0P sample is supplied, uncustomized, in the TDFHINST library and also customized by DFHISTAR, in the XDFHINST library.

## Setting the VTAM APPLID for a CMAS

The last step is to use the CMTCMDEF view to set the VTAM APPLID for the target CMAS, to establish direct CMAS-to-CMAS communication links. The CMTCMDEF view is described in the *CICSplex System Manager Administration* manual.

The CMTCMDEF view is not available until you have a CMAS that can access CICSplex SM itself.

## CMAS journaling

A CICSplex SM address space (CMAS) is capable of producing CICS journal records to track a variety of activities in the CICSplex. These journal records provide an audit trail that can aid in the recovery of data or the reconstruction of events that affected the CICSplex.

A journal record can be written when:

- A definition in the data repository is added, removed, or updated
- An operations action is issued against a MAS
- A real-time analysis event is generated.

The journal records are stored in a 32KB buffer and are flushed to the corresponding log streams when the buffer becomes full or when a normal shutdown of the CICS region is initiated.

To force the buffer to be flushed to a log stream when the CICS region is still active, you could specify the WAIT option on the WRITE JOURNALNAME command using EXEC CICS or the CECI transaction.

To request one or more of the record types, specify the appropriate CICSplex SM system parameters in the startup JCL of a CMAS:

### **JRNLDEFCH(YES)**

For data repository definition changes

### **JRNLOPACT(YES)**

For operations actions

### **JRNLRTAEV(YES)**

For real-time analysis events

For more information on these parameters, see “CICSplex SM system parameters” on page 305.

If you do not want to use the default log stream name of EYUJRNL, you must define a JOURNALMODEL resource in the CSD that has the desired log stream name. To make the JOURNALMODEL resource definition available during CMAS initialization, you must include the JOURNALMODEL resource definition in a CSD group list. This group list should then be included in your CMAS startup using the SIT GRPLIST parameter. To add the JOURNALMODEL resource to the CSD, either edit and run the JCL contained in sample member C1CSTS32.CPSM.SEYUSAMP(EYUJRNE\$) to execute batch utility DFHCSDUP or use the CICS CEDA transaction. You must also update the CICS system initialization (SIT) parameters used to start the CMAS by setting the GRPLIST parameter to reference the new group list.

The journal records produced by a CMAS contain data mapped by a DSECT called EYUBCPJR. Each record consists of a standard prefix and a variable data area. The contents of the data area are specific to the type of journal record being written.

Figure 24 on page 272 shows the format of EYUBCPJR.

```

*-----*
*           EYUBCPJR DSECT Prefix           *
*-----*

```

```

EYUBCPJR      DSECT
EYUBCPJR      DS    0D

CPJR_PREFIX   DS    0D           Prefix of record
CPJR_CMASNAME DS    CL8         CMAS Name which produced record
CPJR_CONTEXT  DS    CL8         Plex Name
CPJR_SCOPE    DS    CL8         Scope Name
CPJR_USER     DS    CL8         User Name
CPJR_STCK     DS    D           Store clock
CPJR_VERSION  DS    H           Current record version
CPJR_VER_ZERO EQU  0000         Version 0
CPJR_VER_ONE  EQU  0001         Version 1
CPJR_VER_CURR EQU  CPJR_VER_ONE Current Version
CPJR_TYPE     DS    H           Record type
CPJR_TYPE_DEFCH EQU  0001         Definition Add/Change/Delete
CPJR_TYPE_RTAEV EQU  0002         Rta Event
CPJR_TYPE_OPACT EQU  0003         Operation action
CPJR_LENGTH   DS    F           Length of entire record plus x
                                prefix area
                                Available for use
CPJR_LEN      DS    FL8
CPJR_DATA_AREA DS    0H         Length of Prefix area
                                Data area

```

```

*-----*
*           Data record for RTA Events       *
*-----*

```

```

CPJR_RTA_DATA DS    0H
CPJR_RTA_TYPE DS    X           Record type
CPJR_RTATYPE_CRT EQU  0001         Event Created
CPJR_RTATYPE_REM EQU  0002         Event Removed
CPJR_RTATYPE_UPD EQU  0003         Event Updated
CPJR_RTATYPE_RES EQU  0004         Event Resolved
CPJR_RTA_GTYPE DS    X           Generated by type
CPJR_RTAGTYPE_SAM EQU  0001         Event produced by Sam
CPJR_RTAGTYPE_APM EQU  0002         Event produced by Apm
CPJR_RTAGTYPE_MRM EQU  0003         Event produced by Mrm
CPJR_RTA_EVENT DS    CL8         Event Name
CPJR_RTA_MSGSTRT DS    CL30        External Entry Message
CPJR_RTA_MSGEND DS    CL30        External Exit Message
CPJR_RTA_EVENTXT DS    CL30        Event Text
CPJR_RTA_SEVERITY DS    CL3        Severity Level
CPJR_RTA_DATA_L EQU  *-CPJR_RTA_DATA Length of the record

```

```

*-----*
*           Data record for Definition changes *
*-----*

```

```

CPJR_DEF_DATA DS    0H
CPJR_DEF_TYPE DS    X           Record type
CPJR_DEFTYPE_ADD EQU  0001         Definition Added
CPJR_DEFTYPE_DEL EQU  0002         Definition Deleted
CPJR_DEFTYPE_UPD EQU  0003         Definition Update
CPJR_DEF_RESERVED DS    X           Reserved
CPJR_DEF_MAJORNAM DS    CL8         Major Name
CPJR_DEF_MAJORID DS    CL8         ADMIN Restype
CPJR_DEF_MAJORVR DS    XL4         Major Version
CPJR_DEF_MAJORVR_NONE EQU  -1           Major Version None
CPJR_DEF_MINORNAM DS    CL8         Minor Name
CPJR_DEF_MINORID DS    CL8         ADMIN Restype
CPJR_DEF_MINORVR DS    XL4         Minor Version
CPJR_DEF_MINORVR_NONE EQU  -1           Minor Version None
CPJR_DEF_SYSID DS    CL8           System Id where change
                                was originated
CPJR_DEF_DATA_L EQU  *-CPJR_DEF_DATA Length of the record

```

```

272-----Installation Guide-----*
*           Data record for Operation commands *
*-----*

```

```

CPJR_OPS_DATA DS    0H

```

For information on writing a program to access and format CICS journal records, refer to the *CICS Customization Guide*.

## Shutting down a CMAS

You can shut down a CMAS using:

- The WUI **Shutdown...** action button
- The COSD transaction.

It is also possible to use the CMASSTOP command of the CODB system-level debugging transaction to shut down the CMAS, but CODB is restricted and should be used only at the request of IBM customer support personnel

**Note:** You should not attempt to:

- Issue the CEMT PERFORM SHUTDOWN command against a CMAS.
- Cancel the CMAS job from MVS

If you take either of these actions, the CMAS cannot shut itself down properly.

### Using the Shutdown button

You can shut down a CMAS using the **Shutdown...** button in three ways:

1.

- From the main menu, click **CICSplex SM operations views > CMASs known to local CMAS** to open the **CMASs known to local CMAS** tabular view.
- Click the record check box beside the CMAS and click the **Shutdown...** button. The **Shutdown** confirmation view is displayed.
- Click **Yes** to confirm. The **CMASs known to local CMAS** tabular view is redisplayed, showing a status of INACTIVE for that CMAS.

2.

- From the main menu, click **CICSplex SM operations views > CMASs known to local CMAS** to open the **CMASs known to local CMAS** tabular view.
- Click on the CMAS name to display the **CMASs known to local CMAS** detailed view.
- Click the **Shutdown...** button. The **Shutdown** confirmation view is displayed.
- Click **Yes** to confirm. The **CMASs known to local CMAS** tabular view is redisplayed, showing a status of INACTIVE for that CMAS.

3.

- From the main menu, click **CICSplex SM operations views > MASs known to CICSplex** to display the **MASs known to CICSplex** tabular view.
- Click on the CMAS name to display the **CMAS detail** view.
- Click the **Shutdown...** button. The **Shutdown** confirmation view is displayed.
- Click **Yes** to confirm. The **MASs known to CICSplex** tabular view is redisplayed.

### Using the COSD transaction

You can issue from any terminal, including an MVS console, the transaction id:

COSD

## Restarting a CMAS

A CMAS that was shut down normally (using the COSD transaction or WUI SHUTDOWN button ) can usually be restarted with a SIT parameter of START=AUTO. However, you must specify START=COLD if you have:

- Modified any of the CICS resource definitions that are used by the CMAS.
- Added or removed CMAS-to-CMAS (CMTCMDEF) connection definitions.

If a CMAS terminates abnormally (that is, through any means other than the COSD transaction or WUI SHUTDOWN button), you must perform an emergency restart to allow CICS to perform backout processing. You can accomplish an emergency restart of a CMAS in one of two ways:

- If the CMAS is registered with the MVS automatic restart manager (ARM), an emergency restart occurs automatically.
- If the CMAS is not registered with ARM, specify START=AUTO in the CMAS CICS system initialization table (SIT) override parameters.

A CMAS should initialize and function properly after an emergency restart, provided you have made no changes to the CICS resource definitions or CICSplex SM connection definitions.

If you have made any such changes since the last run of the CMAS (that is, the one that terminated abnormally), the CMAS may not function properly. In that case, you should shut down the CMAS using the COSD transaction or WUI SHUTDOWN button and restart it, specifying START=COLD. For information about shutting down a CMAS, see “Shutting down a CMAS” on page 273.

---

## Setting up a CICS managed application system (MAS)

This chapter describes the steps you must perform so that a CICS TS system can be known as a managed application system (MAS) to CICSplex SM. (Throughout the rest of this chapter, a CICS TS MAS is referred to as an MVS MAS.) The following levels of CICS under MVS can connect directly to, and be managed by, CICSplex SM:

- CICS Transaction Server for z/OS, Version 3 Release 1
- CICS Transaction Server for z/OS, Version 2 Release 3
- CICS Transaction Server for z/OS, Version 2 Release 2
- CICS Transaction Server for OS/390, Version 1 Release 3

The information you need is in the following sections:

- “Before you begin”
- “Using CICS global user exits and user-replaceable modules” on page 275
- “Controlling the use of modules from the LPA” on page 275
- “Preparing a MVS MAS” on page 276
- “Stopping and restarting management of a CICS system” on page 283.

For a summary of the MAS setup tasks that you can refer to while performing them, see “CICSplex SM setup checklist and worksheets” on page 9.

### Before you begin

Before you begin, check the IEASYSxx member of SYS1.PARMLIB that you use for MVS initialization and make note of the initialization values that are referenced during installation. For details about these values, see “Noting IEASYSxx values for CICSplex SM” on page 100.



If you are converting your CICSplex SM system or systems from a previous release to CICSplex SM for CICS TS for z/OS, Version 3.2, you should read the *CICS Transaction Server for z/OS Migration from CICS TS Version 3.1* or equivalent for your level of CICSplex SM.

For details on applying corrective or preventive maintenance to CICSplex SM, see Chapter 29, “Applying service to CICS Transaction Server for z/OS,” on page 207.

## Using CICS global user exits and user-replaceable modules

This section describes the CICS global user exits (GLUE) and user replaceable modules that are used by CICSplex SM.

The way these exits are used by CICSplex SM conforms to the standard described in the *CICS Customization Guide*. CICSplex SM uses these exits only to acquire information; the application environment is not altered.

CICSplex SM uses the dynamic routing program user replaceable module (DTRPROG) as part of workload balancing.

The XMNOUT and XSTOUT exits are used when monitoring services are enabled for a managed application system (MAS).

- The XMNOUT exit is used to get task and CICS monitoring data. XMNOUT is used only with a local MAS.
- The XSTOUT exit is used to get statistical data before the data is reset by CICS.

These exits are used to obtain monitoring and statistics information and always return a “continue processing” return code. They are disabled when a shutdown request for the MAS is received.

The XMEOUT, XDUREQ, XDUREQC, XRSINDI and XDUOUT exits are used when topology requests are enabled for a local MAS. The XMEOUT exit is used to detect short on storage sick and well health events.

- The XRSINDI exit is used to detect topology resource changes.
- The XDUREQ exit is used to detect system dump and transaction dump sick health events.
- The XDUREQC exit is used to detect the completion of dump action.
- The XDUOUT exit is used to detect transaction dump well health events.
- The XSNOFF exit is used to detect user signoff events.

## Controlling the use of modules from the LPA

You can control whether CICS uses modules from the LPA, by specifying the LPA and PRVMOD CICS system initialization parameters or by including or excluding the SYS1.CICSTS32.CPSM.SEYULPA library (defined to MVS as an LPA library) in the STEPLIB or DFHRPL concatenations.

### Note:

1. A module that is link-edited with the RMODE(ANY) attribute is loaded into the ELPA.
2. It is important to remember that the LPA-resident version of a module usually loaded from STEPLIB will not be used from the LPA if it is left in the STEPLIB DD concatenation of libraries. If a module is found in the STEPLIB concatenation, it is loaded into the private area of the address space, and the LPA version ignored. This situation can be avoided by

moving the LPA-eligible modules into an LPA library, as described in “Installing CICSplex SM modules into the LPA” on page 141.

For further information about controlling the use of LPA-eligible modules, see Chapter 16, “Installing CICSplex SM modules in the MVS link pack area,” on page 141, taking particular note of information concerning:

- The module-not-found warning message (DFHLD0109I)
- CICS SIT parameters related to LPA modules.

## Preparing a MVS MAS

**Note:** Because a CICS system is unknown to CICSplex SM until the CMAS with which the CICS system is associated is started, you should start the CMAS before any of the MASs (that is, the CICS systems the CMAS is to manage).

CMAS links can fail if you have high priority jobs running alongside your CICS regions. For example, if a MAS holding a lock cannot be dispatched because of a priority conflict, the CMAS links can fail.

### MAS data set creation and customization

DFHISTAR allows you to create and customize your managed CICS system (MAS) data sets, according to the parameters you set when you submit the DFHISTAR job.

### DFHISTAR post-installation members for a MAS

When you run DFHISTAR, with a SCOPE of POST or ALL, it creates the following post-installation members for a managed CICS system (MAS) in the XDFHINST library:

- EYUCSYDS – creates and initializes all the data sets for a MAS. EYUCSYDS includes steps to delete the data sets so that you can rerun the job, if required. These deletions are expected to fail the first time you run the job. EYUCSYDS contains the following steps:
  - DELHIST and DEFHIST delete and define CICSplex SM history data sets EYUHISTA and EYUHISTB.
  - HISTINIT uses the EYU9XHID utility to initialize the history data sets.
  - JES3DELA and JES3DEFA are included if you specify the DFHISTAR JES=JES3 option. They delete and define the CICS local catalog, the global catalog and the local request queue.
  - DELREGDS deletes the CICS data sets.
  - DEFTRACE defines the CICS auxiliary trace data sets, DFHAUXT and DFHBUXT.
  - DEFHTML defines the CICS DFHHTML data set.
  - DEFDMPS defines the CICS transaction dump data sets, DFHDMPA and DFHDMPB.
  - DEFTSTD defines the CICS auxiliary temporary storage data set, DFHTEMP.
  - DEFINTD defines the CICS intrapartition transient data set, DFHINTRA.
  - DEFLCD defines the CICS local catalog, DFHLCD.
  - INITLCD uses the DFHCCUTL utility to initialize the CICS local catalog.
  - DEFGCD defines the CICS global catalog, DFHGCD.
  - INITGCD uses the DFHRMUTL utility to initialize the CICS global catalog.

- DEFLRQ defines the CICS local request queue data set, DFHLRQ.
- JES3DELA and JES3DELB are included if you specify the DFHISTAR JES=JES3 option. They delete and define the CICS DFHCSD data set.
- DELCSD deletes the CICS DFHCSD data set.
- DEFCSO defines the CICS DFHCSD data set.
- INITCSD uses the DFHCSDUP utility to initialize the DFHCSD data set.
- EYUJHIST – JCL to create a pair of CICSplex SM history data sets. EYUCSYDS includes steps to create two history data sets, EYUHISTA and EYUHISTB. You can use EYUJHIST if you want to add more history data sets (up to a maximum of 26). It contains the following steps:
  - DELHIST and DEFHIST delete and define a pair of CICSplex SM history data sets.
  - HISTINIT uses the EYU9XHID utility to initialize the history data sets.
- EYULMSSP – CICS SIT overrides for a managed CICS system.
- EYULMSOP – CICSplex SM EYUPARM parameters for a managed CICS system.
- EYUCSYSP – Procedure to start a managed CICS system.
- EYUCSYSJ – JCL to start a managed CICS system. It executes EYUCSYSP.

If you use the default values for the CICSplex SM parameters, the EYUCSYSP PROC statement is as follows:

```
EYUCSYSP PROC DSNCSO='CICSTS32.CPSM.CSYS01.DFHCSD',
              RGNHLQ='CICSTS32.CPSM.CSYS01',
              CICSHLQ='CICSTS32.CICS',
              CPSMHLQ='CICSTS32.CPSM',
              PRMLIB='CICSTS32.XDFHINST',
              CICSPRM=EYULMSSP,           CICS Parameters
              CPSMPRM=EYULMSOP          CPSM Parameters
```

## Customizing post-installation jobs using DFHISTAR

You can use DFHISTAR to generate copies of the managed CICS system post-installation jobs for a different CICS region. DFHISTAR has a SELECT parameter that allows you to specify a new name for a copy of a post-installation job. It has the format:

```
SELECT jobname newname
```

where:

**jobname**

is the name of the job you want to regenerate

**newname**

is the name for the new copy.

You can specify more than one SELECT parameter to select multiple jobs to be regenerated in a single run of the DFHISTAR job. Once you include a SELECT parameter in the DFHISTAR job, only those jobs specified by the SELECT are generated.

For a MAS with the name CSYS02 and a CICS system identifier of CS02 you can change your DFHISTAR parameters to specify the following:

```
CMASNAME name of the CMAS to which this managed system connects
CMSSYSID CICS system identifier of the CMAS to which this managed system connects
CSYSPLEX name of the CICSplex to which this managed system is to be associated
```

```

|           CSYSNAME CSYS02
|           CSYSYSID CS02
|           SELECT EYUCSYDS CS02CSDS           JCL to create the data sets for CSYS02
|           SELECT EYULMSSP CS02CSSP           CICS SIT overrides for CSYS02
|           SELECT EYULMS0P CS02CS0P           CICSplex SM EYUPARM parameters for CSYS02

```

You can then start the managed CICS system CSYS02, using the procedure EYUCSYSP, by specifying:

```

| START EYUCSYSP, DSNCSO='CICSTS32.CPSM.CSYS02.DFHCSO',
| RGNHLQ='CICSTS32.CPSM.CSYS02', CICSJCLQ='CICSTS32.CICS', CPSMHLQ='CICSTS32.CPSM', PRMLIB='CICSTS32.XD
| CICSPRM=CM02CSSP, CPSMPRM=CM02CS0P

```

If you are using EYUCSYSJ to start the WUI, you should edit it to specify the following:

```

| CICSPRM=CS02CSSP, CPSMPRM=CS02CS0P

```

### Preparing to start an MVS MAS

In order for a CICS system to be managed by CICSplex SM, you must:

- Define the system to CICSplex SM, as described in *../com.ibm.cics.ts.doc/eyua7/topics/eyua7eh.dita#eyua7eh* in the *CICSplex System Manager Administration* manual.
- Change the startup JCL for that system by:
  - Modifying the DD statements shown in Figure 25, to include the CICSplex SM data sets
  - Verifying that the appropriate CICS SIT parameters are included

```

|           ...
| //STEPLIB DD DSN=CICSTS32.CPSM.SEYUAUTH,DISP=SHR
| //DFHRPL DD DSN=CICSTS32.CPSM.SEYULOAD,DISP=SHR
| //EYUPARM DD DSN=(Any PO or PS data set with LRECL=80)
| //EYUHISTA DD DSN=(Optional 1st history dataset)
| //EYUHISTB DD DSN=(Optional 2nd history dataset)
| //EYUHISTn DD DSN=(Optional nth history dataset)
|           ...

```

Figure 25. MVS MAS-specific JCL requirements

When changing these DD statements in the startup JCL for a CICS system make sure that the:

#### STEPLIB DD statement

Includes the CICSTS32.CPSM.SEYUAUTH authorized load library.

#### DFHRPL DD statement

Includes the CICSTS32.CPSM.SEYULOAD load library.

#### EYUPARM DD statement

Identifies the library containing the CICSplex SM parameters.

#### Note:

1. Member EYULMS0P in the CICSTS32.CPSM.SEYUPARM data set, contains sample system parameters for a local MAS; this member must be edited. See “CICSplex SM system parameters” on page 305 for a detailed description of each parameter. EYULMS0P is supplied, uncustomized, in TDFHINST and customized in XDFHINST.
2. If you want to use Business Application Services to install CICS resources in a MAS, you must specify the CICSplex SM system

parameter MASPLTWAIT(YES) for that system. This parameter suspends CICS PLT processing until all CICS resources are installed and the CICSplex SM MAS is fully initialized.

#### **EYUHISTx DD statement**

Identifies the history data sets for the MAS. Each MAS must have its own set of CICSplex SM history data sets. The data sets must be allocated to the MAS region by means of DD cards in the JCL with DD names of the form EYUHISTx, where x is a character suffix taking values A through Z. Dynamic allocation is not supported. The data sets must be allocated with a disposition of OLD. The suffix letters must be used in ascending sequence and no letter should be omitted. For example, if four history data sets are required, use DD names EYUHISTA, EYUHISTB, EYUHISTC, and EYUHISTD. See “Preparing the MAS for history recording” on page 282.

### **Activating DB2 connections during CICS startup**

This section applies to CICS Transaction Server for z/OS, Version 3 Release 2 CICS systems with the CICS-DB2 attachment facility.

Special considerations apply when BAS is used to install a DB2 connection defined to CICSplex SM via a DB2CDEF resource definition.

When BAS is used to define and then install a DB2 connection the connection starts out in NOTCONNECTED status. You can use the WUI to cause the connection to the DB2 subsystem to be activated as follows: From the WUI Main menu click CICS operations views > DB2, DBCTL and WebSphere MQ operations views > Connections (under DB2 operations views) > Select the objects to change and click the Connect button.

In a test environment, it may be acceptable to wait for the MAS to start and then install the BAS definition, and issue a CONNECT command against the resulting DB2CONN.

However, in a production system, you may want the connection to be automatically activated when the MAS starts up, as part of the PLT processing sequence, so that the DB2 subsystem can be accessed immediately by programs and users.

Specifying the CICS SIT parameter DB2CONN=YES does not by itself achieve this, because at the time this and other SIT parameters are processed, CICSplex SM has not yet installed any DB2CDEF objects.

The way to activate a DB2 connection during CICS startup is as follows:

1. Ensure there is an appropriate DB2CDEF resource definition for CICSplex SM to install, and that the definition is set up for automatic installation.
2. Specify CICSplex SM parameter MASPLTWAIT(YES). This causes the DB2CDEF resource definition (as well as all other BAS resource definitions) to be installed during PLT processing.
3. Arrange for the appropriate DB2 connect program to be started *after* the MAS startup program (EYU9NXLM for a local MAS).

### **WebSphere MQ connections**

You cannot use BAS to define and install a WebSphere MQ connection before the CICSplex SM environment has been initialised.

## MVS MAS-related CICS SIT parameters

You should verify that the sequential data set or partitioned data set member identified by the CICS SYSIN statement includes the appropriate CICS system initialization table (SIT) parameters, as described in Table 18.

**Note:** To get all data available for the TASK and MLOCTRAN views, MCT must have a value specified, CICS monitoring for performance classes must be activated, and you must be collecting performance class data. See the note for MCT, MONITOR, MN, and MNPER parameters in the following table .

Table 18. CICS SIT parameters for an MVS MAS

Parameter	Explanation
APPLID=	VTAM application ID for this CICS system. Used as MAS name when NAME(va1ue) is not specified as a CICSplex SM system parameter.
AEXIT=DFHZATDX	VTAM terminal autoinstall program.
AUTORESETTIME=YES	Time-of-day synchronization.
AUXTR=ON	Auxiliary trace - exception records.
AUXTRSW=NEXT	No continuous auxiliary trace switching.
CPSMCONN=LMAS	Initialize the region as a local MAS.
DFLTUSER=userid	Specify the user identifier that is to be used for security checking when a user is not defined to the ESM.
DSALIM=5M	Limit of DSA storage below 16MB. Should be set to at least 5MB.
EDSALIM=50M	Limit of EDSA storage above 16MB. Should be set to at least 50MB.
DSRTPGM=EYU9XLOP	Distributed START routing program.
DTRPGM=EYU9XLOP	Dynamic routing program.
DUMPDS=A	Transaction dump data set.
DUMPSW=NEXT	Switch to next transaction dump data set.
FCT=NO	No File control table.
GMTEXT='CICSplex System Manager - CICS Transaction Server for z/OS'	Default logon message.
GRPLIST=DFHLIST	Add group lists for your application resource definitions. (See "CICS resource definitions for CICSplex SM" on page 226 for additional information.)
ICV=100	Region exit interval.
ICVR=5000	Runaway task interval.
ICVTS=1	Terminal scan delay interval.
INTTR=ON	Activate main storage trace.
IRCSTRT=YES	IRC Started at system initialization.
ISC=YES	Code YES to include the CICS programs required for interregion and intersystem communications.
MCT=	Monitoring control table. If you have CICS performance class monitoring active, then you must specify a value for this parameter. You can use 2\$ (the default) or an existing table. (See <b>Note</b> below.)
MN=ON	Activates CICS Monitor. (See <b>Note</b> below.)
MNFREQ=001500	Writes performance class data every 15 minutes.
MNPER=ON	Tells CICS to monitor performance classes. (See <b>Note</b> below.)

Table 18. CICS SIT parameters for an MVS MAS (continued)

Parameter	Explanation
<p><b>Note for MCT, MONITOR, MN, and MNPER parameters:</b> To get all data available for the TASK and MLOCTRAN views, MCT must have a value specified, CICS monitoring for performance classes must be activated, and you must be collecting performance class data.</p> <p>If you do not want this data written to an SMF data set, you can suppress the monitor records. See the description of the SUPPRESSCMF parameter in “CICSplex SM system parameters” on page 305.</p>	
MXT=120	<p>Maximum tasks. Increase by 20 to accommodate the CICSplex SM MAS tasks.</p> <p><b>Note:</b> CICSplex SM rarely uses all 20 of these additional tasks. If you are using the MXT value alone to control application transactions, increasing this value may allow more application transactions to run concurrently. To prevent this from occurring, you can define a transaction class for the application. Then, set a class maximum task (CMXT) value that limits the number of concurrent transactions.</p>
SEC= {YES NO}	<p>Indicate whether external security checking is to be performed for this CICS system. Specify:</p> <p><b>YES</b> When READ access is granted:</p> <ul style="list-style-type: none"> <li>• READ is permitted</li> <li>• UPDATE is refused.</li> </ul> <p>When UPDATE access is granted:</p> <ul style="list-style-type: none"> <li>• READ is permitted</li> <li>• UPDATE is permitted.</li> </ul> <p><b>NO</b> Security checking is not performed.</p> <p><b>Note:</b></p> <ol style="list-style-type: none"> <li>1. For CICS security, the value specified with SEC= for a CMAS overrides the value specified with SEC= for a MAS. (For more information about this parameter, see the <i>CICS RACF Security Guide</i>.)</li> <li>2. For CICSplex SM security to be active, you must set SEC=YES for a MAS, and the CMAS to which it connects must have the CICSplex SM system parameter SEC=YES). When CICSplex SM security is not activated in the CMAS, the connection between the CMAS and the MAS cannot be established. If this is attempted, message EYUCR0007E is issued to the console, the CMAS joblog, and the EYULOG.</li> </ol> <p>(For more information about the SEC parameter for the CMAS, see “CICSplex SM system parameters” on page 305.)</p>
SECPRFX={YES   NO   <u>prefix</u> }	Specify whether the user ID is used as the prefix that is added to the beginning of all resource names to distinguish this CICS system from other CICS systems.
SIT=6\$	System initialization table suffix.
SPOOL=YES	System spooling interface.
START=AUTO	Cold start overriding other options.
SYSIDNT=	Indicate the id of the CICS system. This name should be unique within a CICSplex.
SYSTR=OFF	Auxiliary trace - No system activity.
TCT=NO	No TCT needed.
TS=(COLD,3)	Cold start temporary storage.
TST=NO	No TST needed.

Table 18. CICS SIT parameters for an MVS MAS (continued)

Parameter	Explanation
USERTR=ON	Auxiliary trace - Enable user trace.
XAPPC=NO	RACF checking of APPC sessions.
TCPIP=YES	Activate CICS TCPIP services.
XCMD= { <u>YES</u> lname NO}	Indicate whether EXEC CICS system commands are to be included in security checking. Specify YES or NO. For information about CICSplex SM transaction security requirements, see Defining the CICSplex SM transactions in a MAS in the <i>CICS RACF Security Guide</i> .
XDB2= {No name}	Indicate whether DB2 resources are to be included in security checking. Specify NO or name.
XDCT=NO	RACF checking of DCT entries.
XEJB=NO	RACF checking of security roles.
XFCT= { <u>YES</u> lname NO}	Indicate whether file control entries are to be included in security checking. Specify YES or NO. For information about CICSplex SM transaction security requirements, see Defining the CICSplex SM transactions in a MAS in the <i>CICS RACF Security Guide</i> .
XHFS=NO	Security checking of Web client access to z/OS UNIX files.
XPCT= NO	RACF checking of EXEC-started transactions. For information about CICSplex SM transaction security requirements, see Defining the CICSplex SM transactions in a MAS in the <i>CICS RACF Security Guide</i>
XPPT= { <u>YES</u> lname NO}	Indicate whether program entries are to be included in security checking. Specify YES or NO. For information about CICSplex SM transaction security requirements, see Defining the CICSplex SM transactions in a MAS in the <i>CICS RACF Security Guide</i> .
XRES=NO	Security checking of access to CICS document templates.
XRF=NO	XRF support not generated.
XPSB=NO	RACF checking of PSBs.
XTRAN=NO	RACF checking of transaction-attach.
XTST=NO	RACF checking of TST entries.
XUSER={ <u>YES</u>  NO}	Indicates whether CICS is to perform surrogate user checks. If you specify YES, the CICS region userid must be defined as a surrogate of the userid that starts a MAS using the COLM or CORM transaction.

## Preparing the MAS for history recording

CICSplex SM provides the ability to save and view data for completed tasks, that is historical task data. When an active task completes, its data is stored in a historical data store. The data store is made up of a number of VSAM KSDS data sets. There must be a minimum of two data sets and a maximum of twenty six data sets.

Each MAS must have its own set of CICSplex SM history data sets. The data sets must be allocated to the MAS region by means of DD cards in the JCL with DD names of the form EYUHISTx, where x is a character suffix taking values A through Z. Dynamic allocation is not supported. The data sets must be allocated with a disposition of OLD. The suffix letters must be used in ascending sequence and no letters should be omitted. For example, if four history data sets are required use DD names EYUHISTA, EYUHISTB, EYUHISTC, and EYUHISTD.

The CICSplex SM history data sets must be defined with the REUSE keyword. Task history recording will use the least recently used data set, or when starting for



the first time EYUHISTA. When EYUHISTA becomes full, it will switch to use EYUHISTB and so on in sequence. Each full data set remains open with its data available until the history recorder has filled all data sets and needs to start reusing the data sets. At this time EYUHISTA is set closed, emptied, re-opened, and reused first, followed by EYUHISTB and so on in sequence. If a data set is reused its previous contents are destroyed.

Until the history recorder requires to empty a data set in order to reuse it, the historical task data is available for use. The data is maintained across CMAS and MAS restarts. The history data sets need not be defined as recoverable, as unit of work recoverability is not required. However, the CICSplex SM history recorder facility does require files to be defined as non-recoverable to avoid unnecessary logging in the MAS region.

In addition, the history data sets must not be defined to use VSAM compression. The CPSM history function initializes the data sets in order to calculate how many records will fit in the data set, so that it may safely use sequential writes to the data set, thereby reducing the I/O overhead. Use of VSAM compression will invalidate that calculation and cause data to be lost when the data set becomes full and a data set switch is required.

CICSplex SM provides a tuning aid in the form of a EYUPARM called HISTRECSMSG to determine the optimum size for history data sets. HISTRECSMSG can activate the periodic output of messages detailing how many thousand records have been written to the data set. There is one record for each completed task. Since CICS file control supports extended format KSDS data sets, large history data sets over 4GB in size may be defined. However when considering using very large data sets, take into account that when the CICSplex SM history recorder reuses a data set by emptying it, a large amount of data is lost and not available for subsequent queries. An alternative approach to having a small number of very large data sets, is to spread the data over more data sets. For example, by having twenty five data sets each one capable of holding one hours worth of completed task data, at least one days worth of data can always be maintained. When the oldest data set is reused, only one hours worth of data is lost.

CICSplex SM provides a sample job, EYUJHIST, for defining and initializing two history datasets. This is supplied, uncustomized, in TDFHINST and customized by DFHISTAR in XDFHINST.

## Stopping and restarting management of a CICS system

This section tells you how to:

- Stop management of a CICS system
- Restart management of a CICS system
- Terminate a CICS system.

### Stopping management of a CICS system

To stop the MAS agent code in an active CICS system, either:

- From the WUI Main menu click CICSplex SM operations views > MASs known to CICSplex > select the CICS systems and click the Stop button, or
- Run transaction COSH in the MAS. COSH may be started at a 3270 terminal, at a console, or via ATI.

Stopping the MAS agent prevents CICSplex SM from accessing the MAS until either the CICS system is restarted (see page “Preparing a MVS MAS” on page 276) or the COLM or CORM transaction is issued (see “Restarting management of a CICS system”).

**Note:** When a MAS is active as a CICSplex SM workload management routing region, and the dynamic routing program is set to EYU9XLOP, the STOP command is not honored. In this situation, before you issue the STOP command you must use the WUI **CICS regions** operations view to change the dynamic routing program from EYU9XLOP to the CICS default dynamic routing program, DFHDYP, or another valid dynamic routing program.

### Restarting management of a CICS system

To reactivate a running CICS system as a MAS, issue the CICS transaction:

**COLM** For a local MAS

**Note:** If you want a local MAS to be recognized as a workload management routing region when CICSplex SM resumes managing the system, make sure the dynamic routing program is set to EYU9XLOP. To change the dynamic routing program, use the CICS CEMT transaction before you reactivate the local MAS.

### Terminating a MAS

To verify that the CICSplex SM MAS shutdown processing is properly installed, you can terminate the CICS system and check the log for the following shutdown message:

```
EYUXL0016I MAS shutdown complete
```

To terminate a CICS system running the MAS agent code: From the WUI Main menu click CICS regions > select the CICS system(s) and click the Shutdown button. For more information about the CICS regions views, see the *CICSplex System Manager Operations Views Reference* manual.

---

## Setting up a CICSplex SM Web User Interface server

This chapter describes the steps you must perform to use the CICSplex SM Web User Interface. These steps consist of:

- “Preparing a CICS system to act as the Web User Interface server”
- “Configure CICS Web support” on page 286
- “Review CICS system initialization parameters” on page 286
- “Specify language and codepage” on page 287
- “Prepare codepage conversion table” on page 288
- “Create the Web User Interface server repository (EYUWREP)” on page 288
- “Specify the customizable view and menu help data set” on page 292
- “Specify the Web User Interface server initialization parameters” on page 292
- “Specify the JCL DD statements” on page 301
- “Security considerations” on page 302
- “Additional tasks” on page 304

## Preparing a CICS system to act as the Web User Interface server

You should consider the following before you set up your CICS system to act as your Web User Interface server:

- The CICS system that you select to act as your Web User Interface server must be a dedicated CICS Transaction Server for z/OS, Version 3 Release 2 CICSplex SM MAS connected to a CICS Transaction Server for z/OS, Version 3 Release 2 CMAS. For information about how to set up a MAS, see “Setting up a CICS managed application system (MAS)” on page 274.
- Decide how many Web User Interface servers you need. For example:
  - If you intend to support more than one national language you need a Web User Interface server for every language you want to support.
  - You may want to have Web User Interface servers on multiple MVS images.
  - You may want to have more than one Web User Interface server for availability reasons.
  - The Web User Interface server creates and maintains state data when a user signs on via a web browser (or when an application using the data interface DATA/CONNECTs). Because of this state data, an affinity between the web browser (or application using the data interface) and server is created. The use of techniques like dynamic virtual IP addresses (DVIPA) or distributed DVIPA may not be able to honor this affinity. If this affinity is not honored, the usual result is for web browsers to redisplay the sign on screen or for data interface applications to receive a BADSTUB status.
  - If a the Web User Interface server has a different local IP address or name to the one used by users in their web browsers, for example due to a firewall or another reason that causes network address translation (NAT), the **TCPIPHTTPhost** Web User Interface server initialization parameter may be used to control the way the Web User Interface server generates URLs sent to web browsers.

To set up your CICS system you should:

1. Create the CICS system and confirm that it is operational using the CICS-supplied installation verification procedures (IVPs).
2. Configure CICSplexes for your Web User Interface servers. We suggest that you configure a separate CICSplex for your Web User Interface servers.
3. Ensure that the CMAS to which the Web User Interface connects is managing all CICSplexes that the Web User Interface server needs access to. This is because the Web User Interface server acts as an CICSplex SM API application. However, it is not necessary for the CMAS, to which the Web User Interface connects, to be managing any of the MASs in these CICSplexes. If there is more than one CMAS on the MVS image, that the Web User Interface server will run on, you need to consider which CMAS the Web User Interface should connect to depending on which CICSplexes the CMAS is managing. To control this connection you can either:
  - Ensure the CICSplex to which the Web User Interface server local MAS belongs is only managed by the CMAS or CMASs that the Web User Interface should connect to, or
  - Ensure that the Web User Interface server connects to a specific CMAS by specifying the **CMASYSID EYUPARM** for the server local MAS.
4. Define the Web User Interface server CICS system to CICSplex SM as a local MAS and ensure that the CICS system has been set up correctly using the CICSplex SM installation verification procedures (IVPs).
5. Consider basic monitoring of your Web User Interface servers. You can use standard CICSplex SM monitoring because the Web User Interface server is defined as a MAS.

## Configure CICS Web support

The *CICS Internet Guide* explains how to set up the base components of CICS Web support and verify its operation using the supplied sample programs.

The *CICS RACF Security Guide* explains the additional configuration that is required if you want to use the secure sockets layer (SSL).

### Note:

1. On Web User Interface initialization, a TCPIP SERVICE resource definition is created and opened for you by the Web User Interface. However, you will need to create a temporary TCPIP SERVICE resource definition to run the CICS Web support sample applications. This temporary TCPIP SERVICE resource definition should be discarded after CICS Web support has been tested and before Web User Interface initialization has begun.
2. For SSL, the Web User Interface can either use the default certificate in the key database or a named certificate. However, it can only use a named certificate provided that the label contains only alphanumeric characters and is a maximum of 32 characters.

## Review CICS system initialization parameters

You need to specify the storage key for the CICS common work area (CWA), and the amount of storage required for the CWA on the CWAKEY and WRKAREA CICS system initialization parameters as follows:

```
CWAKEY=CICS  
WRKAREA=2048
```

To ensure that Web User Interface exception trace entries are written to the CICS auxtrace data set, as required to achieve first failure data capture, you should specify the USERTR, SYSTR, and AUXTR CICS system initialization parameters as follows:

```
USERTR=ON  
SYSTR=OFF  
AUXTR=ON
```

**Note:** If you set AUXTR=OFF, this is overridden at startup. The Web User Interface sets auxiliary trace on in order to record exception trace entries in the event of a problem that does not result in a SVCDUMP.

Specify the CPSMCONN CICS system initialization parameter to invoke CICSplex SM code automatically during CICS initialization and initialize the region as a CICSplex SM Web User Interface server. This is the recommended alternative to specifying the CICSplex SM WUI initialization and shutdown programs in initialization and shutdown program list tables (PLTPI and PLTSD).

```
CPSMCONN=WUI
```

You should specify the action CICS should take if, at the next local midnight, the CICS time-of-day differs from the system time-of-day by more than 30 minutes (for example, after setting clocks forward or back to adjust for Summer and Winter time) as follows:

```
AUTORESETTIME=YES
```

Clients should continue to issue the CEMT PERFORM RESET command.

In addition to specifying the necessary CICS system initialization parameters for CICSplex SM local MAS execution, specify the appropriate CICS system initialization parameters to enable CICS Web Interface support, for your release of CICS. For example, TCPIP=YES for CICS Transaction Server for OS/390, Version 1 Release 3 or later.

For information about these parameters see the *CICS System Definition Guide*.

## Specify language and codepage

In addition to specifying CICS system initialization parameters for the CICS Web Interface and CICSplex SM local MAS execution, the Web User Interface requires an INITPARM system initialization parameter to specify the server language and the client codepage. You need to code EYU9VKEC to represent the language of the Web User Interface server and EYU9VWAN to represent the codepage of the client on the INITPARM parameter.

You can select the server language and the client codepage from Table 19 and specify them on the INITPARM parameter as follows:

```
INITPARM=(EYU9VKEC='xxx',EYU9VWAN='yyyy')
```

where xxx is the language identifier of the Web User Interface server and yyyy is the codepage identifier of the client.

Table 19. Language and codepage identifiers for INITPARM

Language	Language identifier (EYU9VKEC)	Client codepage	Default client codepage identifier (EYU9VWAN)
US English	ENU	ISO-8859-1 (819)	ENU1
Japanese	JPN	Shift-JIS (943)	JPN1
Simplified Chinese	CHS	GB2312 (1381)	CHS1
Simplified Chinese	CHS	GB18030 (05488)	CHS2

For example, if your chosen language was English you would code the INITPARM parameter as follows:

```
INITPARM=(EYU9VKEC='ENU',EYU9VWAN='ENU1')
```

### Note:

1. The codepage identifier can be overridden for individual user requests by placing it in the URL used to access the Web User Interface. For example:

```
http://hostname:port/CICSPlexSM/codepage
```

where *hostname* is the name specified on the TCPIPHOSTNAME Web User Interface server initialization parameter and *port* is the value specified on the TCPIPPORT Web User Interface server initialization parameter. For information about the Web User Interface server initialization parameters see “Specify the Web User Interface server initialization parameters” on page 292.

2. If the INITPARM system initialization parameter is not specified or if a value is not specified for EYU9VKEC or EYU9VWAN, the default values are ENU for the language and ENU1 for the codepage. However, operator messages are issued every time default values are used.

3. A simplified Chinese web user interface server can support client browsers using either GB2312 (CHS1) or GB18030 (CHS2). For GB2312 clients, the server uses the EBCDIC codepage, 935. For GB18030 clients, the server assumes a second EBCDIC codepage, 1388, which is a superset of codepage 935. View sets and menus edited using a GB2312 client browser are stored in 935 and can be used in either client codepage. However, if a view set or menu is edited using a GB18030 client and characters not available in 935 are used, the resulting views set or menu are not displayed correctly on GB2312 clients. Note that all supplied messages and starter set views and menus, and view sets and menus customized using CICS TS 2.2 or earlier, use the 935 codepage and can be displayed on either client codepage.
4. Some web browsers do not support all the available client code pages. For example, many older browsers do not support GB18030.

## Prepare codepage conversion table

You can use the default version of DFHCNV, provided in the SDFHLOAD library, which includes the CICSplex SM codepages automatically. If you use your own version of the DFHCNV source module assemble and link-edit it using the CICS procedures for maintaining conversion table load modules.

A sample copybook is provided in CICSTS32.CPSM.SEYUSAMP called EYU\$CNV1, to show what entries are automatically added to DFHCNV when you assemble the table. This copybook contains an entry for every language and client codepage combination that is supported as follows:

### **EYUENU1**

Entry for English

### **EYUJPN1**

Entry for Japanese

### **EYUCHS1**

Entry for simplified Chinese (GB2312 clients).

### **EYUCHS2**

Entry for simplified Chinese (GB18030 clients)

**Note:** It is not necessary to include a copy statement for EYU\$CNV1 in the DFHCNV source.

## Create the Web User Interface server repository (EYUWREP)

The Web User Interface server repository (EYUWREP) contains the Web User Interface server's view and menu definitions. You can use the IDCAMS utility to create a VSAM file for these definitions as follows:

```

DEFINE CLUSTER (
NAME( dsname )
VOLUMES( dsvol )
RECORDS( 5000 5000 )
RECORDSIZE( 8192 32000 )
CONTROLINTERVALSIZE( 8192 )
SPANNED
INDEXED
KEYS( 20 20 )
SHAREOPTIONS( 2 )
)

```

Figure 26. Sample definition to create Web User Interface repository

Each Web User Interface server must have its own Web User Interface server repository that cannot be shared with any other Web User Interface server.

By default, each Web User Interface server has its own Web User Interface server repository that is not shared with any other Web User Interface server.

You should migrate definitions using the import and export functions. For information, see the *Web User Interface administration* topic in the *CICSplex System Manager Web User Interface Guide*.

You should back up the Web User Interface server repository data as the repository is updated whenever changes are made using the View Editor or when definitions are imported using the COVC transaction. You can back up the Web User Interface repository by using IDCAMS, DFSMSdss or equivalent utility.

Sample JCL, EYUJWREP, creates the Web User Interface repository. The sample JCL is provided, uncustomized, in TDFHINST and customized by DFHISTAR in XDFHINST.

### Sharing the Web User Interface server repository (EYUWREP) across servers

The Web User Interface server repository can be shared across WUI servers if it is accessed using VSAM RLS. Follow these steps to allow sharing of the WUI server repository:

1. Add the parameter **RLS=YES** to the DFHSIT table or when you override system initialization parameters.
2. Add the **LOG(UNDO)** parameter to the repository file definition when you use the IDCAMS facility to create the VSAM file.

During initialization, the Web User Interface server determines whether RLS should be enabled for the Web User Interface server repository and sets the appropriate attributes for the EYUWREP file and enqmodel definition.

The Web User Interface server caches objects in memory to improve performance. If a user logged on to one server creates or updates an object using the View Editor, the User Editor, or the Favorites Editor, users logged on to other servers which share the server repository might not see the changes until their local cache is reloaded. Cache is reloaded by restarting the server.

## WUI data set creation and customization

DFHISTAR allows you to create and customize your Web User Interface (WUI) server data sets, according to the parameters you set when you submit the DFHISTAR job.

### DFHISTAR post-installation members for a WUI

When you run DFHISTAR, with a SCOPE of POST or ALL, it creates the following post-installation members for a CICSplex SM Web User Interface (WUI) server in the XDFHINST library:

- EYUWUIDS – creates and initializes all the data sets for a WUI. EYUWUIDS includes steps to delete the data sets so that you can rerun the job, if required. These deletions are expected to fail the first time you run the job. EYUWUIDS contains the following steps:
  - DELWREP deletes the CICSplex SM WUI data repository, EYUWREP.
  - DEFWREP defines the CICSplex SM WUI data repository, EYUWREP.
  - DELCOVDS deletes the WUI import and export data sets, EYUCOVI and EYUCOVE
  - DEFCOVDS defines the WUI import and export data sets, EYUCOVI and EYUCOVE
  - JES3DELA and JES3DEFA are included if you specify the DFHISTAR JES=JES3 option. They delete and define the CICS local catalog, the global catalog and the local request queue.
  - DELREGDS deletes the CICS data sets.
  - DEFTRACE defines the CICS auxiliary trace data sets, DFHAUXT and DFHBUXT.
  - DEFHTML defines the CICS DFHHTML data set.
  - DEFDMPS defines the CICS transaction dump data sets, DFHDMPA and DFHDMPB.
  - DEFTSTD defines the CICS auxiliary temporary storage data set, DFHTEMP.
  - DEFINTD defines the CICS intrapartition transient data set, DFHINTRA.
  - DEFLCD defines the CICS local catalog, DFHLCD.
  - INITLCD uses the DFHCCUTL utility to initialize the CICS local catalog.
  - DEFGCD defines the CICS global catalog, DFHGCD.
  - INITGCD uses the DFHRMUTL utility to initialize the CICS global catalog.
  - DEF LRQ defines the CICS local request queue data set, DFHLRQ.
  - JES3DELA and JES3DELB are included if you specify the DFHISTAR JES=JES3 option. They delete and define the CICS DFHCSD data set.
  - DELCSD deletes the CICS DFHCSD data set.
  - DEFCSD defines the CICS DFHCSD data set.
  - INITCSD uses the DFHCSDUP utility to initialize the DFHCSD data set.
- EYUJWREP – JCL to create the WUI data repository EYUWREP. EYUWUIDS includes steps to delete and define EYUWREP. You can run EYUJWREP if you just want to delete and define EYUWREP but none of the other data sets for a WUI.
  - DELWREP: delete the CICSplex SM WUI data repository, EYUWREP.
  - DEFWREP: define the CICSplex SM WUI data repository, EYUWREP.
- EYUWUIISP – CICS SIT overrides for a WUI.
- EYUWUIOP –CICSplex SM EYUPARM parameters for a WUI.



- EYUWUIIN – CICSplex SM EYUWUI parameters for a WUI.
- EYUWUIP – a procedure to start a WUI.
- EYUWUIJ – JCL to invoke EYUWUIP.

If you use the default values for the CICSplex SM parameters, the EYUWUIP PROC statement is as follows:

```
EYUWUIP PROC DSNCSO='CICSTS32.CPSM.WUINCM01.DFHCSO',
           RGNHLQ='CICSTS32.CPSM.WUINCM01',
           CICSHLQ='CICSTS32.CICS',
           CPSMHLQ='CICSTS32.CPSM',
           PRMLIB='CICSTS32.XDFHINST',
           CICSPRM=EYUWUIISP,           CICS Parameters
           CPSMPRM=EYUWUIOP,          CPSM Parameters
           WUIPRM=EYUWUIIN            WUI Parameters
```

## Customizing post-installation jobs using DFHISTAR

You can use DFHISTAR to generate copies of the WUI post-installation jobs for a different CICS region. DFHISTAR has a SELECT parameter that allows you to specify a new name for a copy of a post-installation job. It has the format:

```
SELECT jobname newname
```

where:

**jobname**

is the name of the job you want to regenerate

**newname**

is the name for the new copy.

You can specify more than one SELECT parameter to select multiple jobs to be regenerated in a single run of the DFHISTAR job. Once you include a SELECT parameter in the DFHISTAR job, only those jobs specified by the SELECT are generated.

For a WUI with the name WUINCM02 and a CICS system identifier of WU02, that connects to a CMAS with the name CMAS02 and CICS system identifier CM02, you can change your DFHISTAR parameters to specify the following:

```
CMASNAME CMAS02
CMSSYSID CM02
WUIPLEX WUIPCM02
WUINAME WUINCM02
WUISYSID WU02
TCPIPHST TCP/IP host name for this WUI
TCPIPPRT TCP/IP port number for this WUI
SELECT EYUWUIDS WU02WUDS           JCL to create the data sets for WUINCM02
SELECT EYUWUIISP WU02WUSP          CICS SIT overrides for WUINCM02
SELECT EYUWUIOP WU02WUOP           CICSplex SM EYUPARM parameters for WUINCM02
SELECT EYUWUIIN WU02WUIN           CICSplex SM EYUWUI parameters for WUINCM02
SELECT EYUWUIJ CW02WUIJ            JCL statement
SELECT EYUWUIP CW02WUIP            PROC statement
```

You can then start the WUI WUINCM02, using the procedure EYUWUIP, by specifying:

```
START EYUWUIP, DSNCSO='CICSTS32.CPSM.WUINCM02.DFHCSO',
           RGNHLQ='CICSTS32.CPSM.WUINCM02', CICSHLQ='CICSTS32.CICS',
           CPSMHLQ='CICSTS32.CPSM', PRMLIB='CICSTS32', CICSPRM=WU02WUSP, CPSMPRM=WU02WUOP,
           WUIPRM=WU02WUIN
```

If you are using EYUWUIJ to start the WUI, you should edit it to specify the following:

CICSPRM=WU02WUSP, CSMPRM=WU02WU0P, WUIPRM=WU02WUIN

## Specify the customizable view and menu help data set

The Web User Interface allows a site to provide customized help for individual views and menus. This help takes the form of HTML documents that can be served by the Web User Interface server or by an external server. If the Web User Interface is to serve the HTML documents, you must provide a partitioned data set to contain the HTML documents.

You can use the View Editor to customize your views and menus to include a link to the customizable view and menu help data set. The View Editor gives you the option of specifying:

- No help to be available for this view or menu
- Member name of a help page to be served by the Web User Interface from DFHHTML
- URL of an external help page to be served by another HTTP server

For information about the View Editor, see the *CICSplex System Manager Web User Interface Guide*.

The Web User Interface server uses the CICS Web Interface template manager to serve the customized view and menu help. For more information, see the *CICS Internet Guide*.

**Note:** The customizable view and menu help data set (DFHHTML) must be a single data set and should not be concatenated with any other data set.

## Specify the Web User Interface server initialization parameters

You can specify the Web User Interface server initialization parameters in the start-up job or in a fixed block 80 data set. See “Specify the JCL DD statements” on page 301 for the DDname. All of these parameters are subject to the following conditions unless otherwise stated:

- Lines with an asterisk in column 1 are comments and are ignored.
- Values must not contain lower case characters.
- Values must be specified in parenthesis immediately following the parameter.
- Values must not be greater than 32 characters.

For example:

```
* An EYUWUI parameter data set
DEFAULTMENU(OURHOME)
TCPIPHOSTNAME(MVSXX.COMPANY.COM)
TCPIPPORT(4445)
```

A description of the Web User Interface server initialization parameters follows with default values for the parameters underlined>.

### Required parameters

The Web User Interface server has some required initialization parameters. If you do not specify a required parameter the Web User Interface server initialization fails. The Web User Interface server initialization also fails if any of the specified parameters are invalid.

The required Web User Interface server initialization parameters are:

**TCPIPHOSTNAME(name)**

Specify the TCP/IP host name of this Web User Interface server. This is normally the host name and domain name of the MVS system (that is, a fully-qualified name). This host name is normally used by the Web User Interface to construct URLs, depending on the client's HTTP version and the value of the TCPIPHTTPhost Web User Interface server parameter.

This value is always returned in the TCPIPHOSTNAME header of a DATA/CONNECT Web User Interface Data Interface request.

The Web User Interface does not support names longer than 32 characters. If required, you may use the address of the server using the dotted decimal notation. For example, '127.0.0.1'.

**TCPIPport(value)**

Specify the TCP/IP port number of the port that you have allocated for the Web User Interface to run on.

**Optional parameters**

You might also want to consider the following Web User Interface server initialization parameters:

**AUTOREFRESH(YES | NO)**

Disable the automatic refresh option for a WUI server. When YES is specified (the default), the automatic refresh control is displayed, based on the view definition. When NO is specified, the automatic refresh control is not displayed, even if automatic refresh is set on the view definition.

**RESOURCELIMIT(WARNING | FAIL)**

Specifies the action when the default warning count threshold is reached. When WARNING is specified (the default), message EYUVC1258W is issued and the user can select OK to ignore the warning threshold. When FAIL is specified, message EYUVC1267E is issued and the resource request is denied. The filter can be changed, but users of the WUI server will not be able to click OK to bypass a warning count.

**Additional TCP/IP parameters:**

**TCPIPADDRESS(name | INADDR\_ANY)**

Specifies the dotted decimal IP address on which the Web User Interface will listen for incoming requests. If name is specified, it must be of the form nnn.nnn.nnn.nnn where nnn is 0 through 255. If INADDR\_ANY is specified (the default), the Web User Interface will listen on any of the addresses known to the TCP/IP for z/OS host.

You do not normally need to specify the TCPIPADDRESS option unless the z/OS host has multiple TCP/IP addresses.

**TCPIPHTTPhost(NO|YES)**

Indicate whether you require the TCP/IP host name used to construct URLs to be generated based on the incoming HTTP request for HTTP version 1.1 requests or later.

This option has no effect on pre HTTP 1.1 requests sent to the Web User Interface server. The Web User Interface server will always construct URLs using the host name specified in the TCPIPHOSTNAME Web User Interface server parameter for HTTP 1.0 (and earlier) requests.

**NO** For HTTP 1.1 (or later) requests, the host name used in URLs

constructed by the Web User Interface server is based on the value specified in the TCPIPHOSTNAME Web User Interface server parameter.

**YES** For HTTP 1.1 (or later) requests, the host name used in URLs constructed by the Web User Interface server is based on the incoming URI or HTTP 'Host' header, according to the HTTP 1.1 specification.

When HTTP 1.1 clients are used with a Web User Interface server running TCPIPHHTTPHOST(YES), the IP address or name used on the server, does not need to be the same as that used by the HTTP 1.1 client. This can allow the Web User Interface to be used when name address translation (NAT) is performed (for example due to a firewall).

If TCPIPSSL(YES) is used with TCPIPHHTTPHOST(YES) and HTTP 1.1 clients are used along with different IP address names, you may receive SSL certificate warnings due to host name mismatches.

#### **TCPIPSSL(YES | NO)**

Indicates whether you require data encryption between your Web User Interface server and web browser. If you select YES, you must have specified the appropriate system initialization parameters to enable SSL support in the CICS Web Interface. For information see the *CICS Internet Guide*.

#### **TCPIPSSLCERT(name)**

Specify (in upper case characters) the label for the SSL certificate that is to be used for the connection between the Web User Interface and the web browser. If you specify an explicit certificate, the label must be no longer than 32 characters.

The default is the default certificate in the key database or key ring, as applicable.

**Note:** The names of all SSL certificates to be used with the WUI server must be upper case.

#### **TCPIPSSLCIPHERS(cipher\_list)**

Specifies a string of up to 44 hexadecimal digits that is interpreted as a list of up to 22 2-digit cipher suite codes. The cipher suite codes are used to describe the set of algorithms that are supported by the Web User Interface server for SSL communications. If the TCPIPSSLCIPHERS parameter is not specified, the cipher suite is set using the ENCRYPTION system initialization option that the Web User Interface server is using. For more details about specifying cipher suites, see the *CICS RACF Security Guide*.

**Import options:** You can specify the following parameters if you want the Web User Interface server to import automatically WUI data repository definitions, which consist of:

- View sets and menus
- WUI map objects
- User objects
- User group profiles

Definitions can be imported from a specified transient data queue or a data set, or both.

Auto-import is an alternative to the import function of the COVC transaction. When using auto-import, you can use all three parameters to specify a transient data queue and a data set. When using COVC, you are restricted to specifying either a transient data queue or a data set in one import operation.

**AUTOIMPORTDSN**(*dsn\_name*)

Specifies the name of the data set containing IBM-supplied view and menu definitions. The data set cannot be longer than 31 characters. Currently, the supplied set of WUI view and menu definitions is in the SEYUVIEW data set. If you specify an AUTOIMPORTDSN name, you must specify the name of a data set member using the AUTOIMPORTMEM parameter.

Use the AUTOIMPORTDSN and AUTOIMPORTMEM parameters when you want to import specific IBM-supplied view set and menu definitions as a result of service (by a PTF).

**AUTOIMPORTMEM**(*member\_name*)

Specifies the name of the data set member containing the specific IBM-supplied view and menu definitions that you want to import. You can use an asterisk at the end of the name to specify a group of data set members that begin with the same characters. For example, specifying AUTOIMPORTMEM(EYUEA\*) with the IBM-supplied SEYUVIEW data set in AUTOIMPORTDSN imports all of the members beginning with the characters EYUEA.

Use the AUTOIMPORTDSN and AUTOIMPORTMEM parameters when you want to import specific IBM-supplied view set and menu definitions as a result of service (by a PTF).

**AUTOIMPORTTDQ**(*tdq\_name*)

Specifies the name of the CICS extrapartition transient data queue from which you want the server to import a complete set of WUI data repository definitions during server initialization. To use this option, you enter a value explicitly. There is no automatic default; however, queue name COVI (DD name EYUCOVI) is defined as shown in sample EYU£WDEF.

You use the AUTOIMPORTTDQ parameter when starting a WUI server for the first time or if you are already a CICSplex SM user and want to import other data repository definitions, including your own customized definitions, into an existing data set. Avoid using this parameter at other times because of the resources required to perform the import operation.

**Data formatting options:** These options determine how data will appear on Web User Interface displays.

**CVDASTYLE**(MIXED | UPPER)

Indicate whether the CVDAs, EYUDAs, and so on should be displayed in upper case or mixed case characters.

**MIXED**

Mixed case text, that is, the first character upper case and the rest lower case, for example, 'Enabled'.

**UPPER**

Text is displayed in upper case only.

**DATEFORMAT** (**format**)

Specify the format to be used to display the date on Web User Interface displays as follows:

**YYMMDD****DDMMYY****MMDDYY****YYYYMMDD****DDMMYYYY**

## MMDDYYYY

where:

**DD** is the day.

**MM** is the month.

## YY and YYYY

are the year in two-digit or four-digit format, respectively.

## DATESEPARATOR(character | / )

Specify the character to be used to separate the date elements on Web User Interface displays.

## DECIMALSEPARATOR(character | . )

Specify the character to be used to denote the decimal point on Web User Interface displays.

## GMMTEXTMSG( NO | YES | BEFORE | AFTER)

Specify how the CICS "good morning" message is handled.

**NO** The message is not issued

**YES** The message is issued before and after signon.

### **BEFORE**

The message is issued before signon only.

### **AFTER**

The message is issued after signon only.

## MSGCASE ( MIXED | UPPER)

Indicate whether messages destined for the operator or EYULOGs should be displayed in mixed case or upper case characters.

### **MIXED**

Mixed case text is displayed.

If you specify MIXED, output may be displayed incorrectly on Katakana display terminals, where lower case characters are displayed as Katakana symbols.

### **UPPER**

Text is displayed in upper case only.

## THOUSNDSEPARATOR(character | , )

Specify the character to be used to separate thousands on Web User Interface displays, when required. For example, 100000 is displayed as 100,000 if the default is used.

### **Note:**

1. The space character (hex 40) is a valid THOUSNDSEPARATOR value, allowing digits to be grouped by a space.
2. Use 0 (zero) to suppress the THOUSNDSEPARATOR value.
3. The THOUSNDSEPARATOR value is used only when required by the individual view definition.

## TIMESEPARATOR(character | : )

Specify the character to be used to separate hours, minutes, and seconds on Web User Interface displays.

**Environment options:** These specify the context and scope values, the home menu and navigation frame, default map objects, and the appearance of newly opened maps.

**AUTOREFRESH**(YES | NO)

Disable the automatic refresh option for a WUI server. The default setting, YES, displays automatic refresh control, based on the view definition. When NO is specified, automatic refresh control is not displayed, even if automatic refresh control is set on the view definition. See Automatic refresh for more information on the automatic refresh option.

**DEFAULTCMASCTXT**(name | EYUCMS1A)

Specify the CMAS context that is set when the user signs onto the Web User Interface.

**DEFAULTCONTEXT**(name | EYUPLX01)

Specify the context that is set when the user signs onto the Web User Interface.

**DEFAULTMAPBAS**(name | EYUSTARTMAPBAS)

Specify the name of the map object used to generate maps of business application services definitions.

**DEFAULTMAPCOLL**(value | 0)

Specify the number of rows in a generated map below which a map opens in the expanded state. If the number of rows to be displayed is above this number, the map opens in a fully collapsed state. The default value of 0 means that in every generated map all of the rows are visible when opened.

**DEFAULTMAPMON**(name | EYUSTARTMAPMON)

Specify the name of the map object used to generate maps of monitoring definitions.

**DEFAULTMAPRTA**(name | EYUSTARTMAPRTA)

Specify the name of the map object used to generate maps of real-time-analysis definitions.

**DEFAULTMAPWLM**(name | EYUSTARTMAPWLM)

Specify the name of the map object used to generate maps of workload management definitions.

**DEFAULTMENU**(name | EYUSTARTMENU)

Specify the name of the menu that is presented to users after sign onto the Web User Interface.

**DEFAULTNAVIGATE**(name | EYUSTARTNAVIGATE)

Specify the name of the navigation frame that is presented to users after sign onto the Web User Interface.

**DEFAULTSCOPE**(name | EYUPLX01)

Specify the scope that is set when the user signs onto the Web User Interface.

**DEFAULTWARNCNT**(value)

Specify the number of records required to trigger the record count warning mechanism before opening a view. This can take an integer value in the range of 0 to 99999999. The default value is 0 meaning no warnings are issued.

**Note:** You can also set a record count warning value applying to a group of users when setting up a WUI user group (this applies only if the WUI is running with security switched on). A value set in a user group takes precedence over a value set in the DEFAULTWARNCNT parameter for the users in that group.

**GLOBALPREFILTER(YES|NO)**

Specify whether users of the Web User Interface are required to specify filter parameters the first time a view is displayed (before data is collected).

**Note:** You can also specify this parameter when setting up a WUI user group (this applies only if the WUI is running with security switched on). A value set in a user group takes precedence (for users in that group) over the value set in the GLOBALPREFILTER parameter.

**RESOURCELIMIT(WARNING | FAIL)**

Specify whether a warning or failure is issued when the resource limit is reached. The default setting, WARNING, issues message EYUVC1258W and the user can select OK to bypass the warning threshold. Specify FAIL to issue message EYUVC1267E and to deny the new resource request. The WUI server can be used to change the RESOURCELIMIT filter, however the WUI server cannot be used to bypass the warning threshold.

**Operation options:** These options name the default view set to be used if the Web User Interface receives an external request that does not specify a view set name but specifies an object name. The view sets that you name in these options must represent the objects that may be specified. For more information see the *Providing access to WUI views and menus* topic in the *CICSplex System Manager Web User Interface Guide*.

These parameters can be ignored if you do not intend to launch Web User Interface displays in this manner.

**DEFAULTCICSplex(name | EYUSTARTCICSplex)**

Specify the name of the default CICSplex view set.

**DEFAULTCICSrgn(name | EYUSTARTCICSrgn)**

Specify the name of the default CICS region view set.

**DEFAULTCONNECT(name | EYUSTARTCONNECT)**

Specify the name of the default connection view set.

**DEFAULTCSysgrp(name | EYUSTARTCSysgrp)**

Specify the name of the default CICS system group view set.

**DEFAULTDB2SS(name | EYUSTARTDB2SS)**

Specify the name of the default DB2 subsystem view set.

**DEFAULTEJCbean(name | EYUSTARTEJCbean)**

Specify the name of the default Enterprise Bean in a CorbaServer view set.

**DEFAULTEJDJbean(name | EYUSTARTEJDJbean)**

Specify the name of the default Enterprise Bean in a CICS-deployed JAR file view set.

**DEFAULTEVENT(name | EYUSTARTEVENT)**

Specify the name of the default event view set.

**DEFAULTLOCFILE(name | EYUSTARTLOCFILE)**

Specify the name of the default local file view set.

**DEFAULTLOCTRAN(name | EYUSTARTLOCTRAN)**

Specify the name of the default local transaction view set.

**DEFAULTPROGRAM(name | EYUSTARTPROGRAM)**

Specify the name of the default program view set.

**DEFAULTREMFIL(name | EYUSTARTREMFIL)**

Specify the name of the default remote file view set.



**DEFAULTRETRAN(name | EYUSTARTRETRAN)**

Specify the name of the default remote transaction view set.

**DEFAULTTASK(name | EYUSTARTTASK)**

Specify the name of the default task view set.

**User options:****INACTIVETIMEOUT(value | 30)**

Specify the period, in minutes, after which inactive user sessions are terminated. The maximum period allowed is 10080 minutes (7 days).

**MAXUSERS(value | 20)**

Specify the maximum number of concurrent users of the Web User Interface. The maximum number of concurrent users allowed is 50.

**SIGNONPANEL(BASIC | ENHANCED)**

Specifies, if the Web user Interface server has CICS security active (SEC=YES in the system initialization parameter), whether the Web User Interface sign on panel takes one of the following actions:

- Displays a GROUP option
- Saves previously used USER and GROUP values
- Positions the cursor in the sign on field requiring input

When the default value, BASIC, is specified, the GROUP option is not displayed on the sign on screen and the user group profile is set to the default group of the user. Values are not saved and the cursor is not positioned on the sign on panel.

When the value, ENHANCED, is specified, the GROUP option is displayed on the sign on screen and the following events occur:

- If the user enters a value, sign on proceeds and if:
  - The user ID is connected to the specified group, the group is used for the WUI user group profile
  - The user ID is not connected to the specified group or the group name is not valid, sign on continues, but the WUI group profile is set to the default group of the user, and message EYUVC1227W is issued to the WUI user after sign on is complete. Message EYUVS0024W is written to the WUI server's EYULOG.
- If the user does not enter a value, the user group profile is set to the default group of the user

The GROUP option does not change the current connect group of the user being signed on. The security environment built by the WUI is always based on the default group of the user. The group option on the WUI signon screen specifies which WUI user group profile the user should be associated with and does not change any security decisions that might be made by the External Security Manager.

When the SIGNONPANEL(ENHANCED) option is set and JavaScript is enabled in the web browser, the sign on process performs the following actions:

- Saves the user ID and group values in a cookie, so that when the signon process next runs, the form is filled with previously entered values
- Positions the cursor to the sign on field requiring input

Whether the Web User Interface user group profile is set based on the default group, or specified on the GROUP option, if no matching Web User Interface group profile is found, the values usually set using a user group profile are set to the system default.

The SIGNONPANEL option is ignored if the Web User Interface server is running with CICS Security inactive (SEC=NO in the system initialization parameter).

**Accessibility options:** These parameters specify the default colors for the Web User Interface displays.

These parameters should not normally be changed as they effect all users of the Web User Interface server. If you want to change these parameters for reasons of accessibility, take care to ensure that the Web User Interface displays do not become unreadable.

Each parameter specifies a color as six hexadecimal digits. Each pair of digits describes the red, green and blue components of the color, respectively. For example, FFFFFFFF represents white, 000000 represents black, FF0000 represents bright red, 00FF00 represents bright green, and 0000FF represents bright blue.

**COLORPAPER(color)**

Main work frame background color.

**COLORPAPERHEAVY(color)**

Navigation and assistance frame background color.

**COLORPAPERLIGHT(color)**

Background color used for many interface items (for example, information messages, table column headings, detail view labels, view selection and refresh area).

**COLORPAPERWARN(color)**

Background color for warning messages.

**COLORPAPERERROR(color)**

Background color for error messages.

**COLORPAPERALT(color)**

Background color for alternate rows on tabular displays.

**COLORPAPERRULE(color)**

Background color for assistance frame bar containing the navigation and help icons.

**COLORINK(color)**

Main work frame text color.

**COLORINKBANNER(color)**

Navigation and assistance frame text color.

**COLORINKLINK(color)**

Unvisited link text color.

**COLORINKVLINK(color)**

Visited link text color.

**Problem determination option:**

**WUITRACE(trace levels)**

Specifies the level of tracing for the Web User Interface server.

The trace levels that you specify must be separated by a comma. For example:  
WUITRACE(8,11,13,15,18)

You can define a range of trace levels, for example:

```
WUITRACE(1:5)
    activates trace levels 1 through 5
```

```
WUITRACE(1:5,13,28:31)
    activates trace levels 1 through 5, 13, and 28 through 31
```

**Attention:** It is recommended that you only activate trace at the request of IBM Support Center personnel.

## Transient data queue definitions

You need definitions for the following transient data queues (TDQs):

**COVP** The Web User Interface server initialization parameters data set, EYUWUI. This is a fixed block 80 input data set.

**COLG** The CICSplex SM output log, EYULOG. This is a variable length output data set.

**COVI** Sample definition for the Web User Interface import data set, EYUCOVI. This is a variable length input data set.

**COVE** Sample definition for the Web User Interface export data set, EYUCOVE. This is a variable length output data set.

### CICS Transaction Server definitions

If you are using the CICSplex SM element of CICS Transaction Server for z/OS, the transient data queue (TDQ) definitions are provided in the EYU\$WDEF sample, which is supplied in the SEYUSAMP library.

COVI and COVE are provided as samples that can be used to create additional import and export transient data queues.

For more information about the COVI and COVE TDQs see the *CICSplex System Manager Web User Interface Guide*.

## Temporary storage models

TSMODELs and TSMDEFs should not be installed into a Web User Interface (WUI) server.

## Specify the JCL DD statements

Additional DD statements should be added to the Web User Interface server for the following data sets:

### EYUWUI

The Web User Interface server initialization parameters data set. See “Specify the Web User Interface server initialization parameters” on page 292.

### EYUWREP

The Web User Interface server repository data set. See “Create the Web User Interface server repository (EYUWREP)” on page 288.

### EYULOG

The CICSplex SM output log.

## DFHHTML

The customizable view and menu help data set. See “Specify the customizable view and menu help data set” on page 292. This is optional.

## EYUCOVI

The Web User Interface server import data set. For more information see “Transient data queue definitions” on page 301 and the *CICSplex System Manager Web User Interface Guide*. This is optional.

## EYUCOVE

The Web User Interface server export data set. For more information see “Transient data queue definitions” on page 301 and the *CICSplex System Manager Web User Interface Guide*. This is optional.

For example,

```
/*  
//DFHHTML DD DISP=SHR,DSN=data set name  
//EYUWREP DD DISP=SHR,DSN=data set name  
//EYUCOVI DD DISP=SHR,DSN=data set name  
//EYULOG DD SYSOUT=*  
//EYUWUI DD *  
DEFAULTMENU(OURHOME)  
TCPHOSTNAME(MVSXX.COMPANY>COM)  
TCPPIPPORT(4445)  
/*
```

## Security considerations

This section describes the Web User Interface security requirements for CICS security, Secure Sockets Layer (SSL) support, and access to MVS data sets.

### CICS security considerations

If your Web User Interface server region is running with CICS security active, you need to define the security access required:

- For the CICS Web Interface
- By the administrator
- By the end-user and users of the View Editor

Table 20 on page 303 summarizes the access required by the various userids.

You may wish to use CICS transaction security (see the *CICS RACF Security Guide*) to limit the users who are allowed to control the Web User Interface server via the COVC transaction.

See the *CICSplex System Manager Web User Interface Guide* for information about how to control users of the Web User Interface and to limit what resources they are allowed to access.

**Security access for the CICS Web Interface:** If CICS transaction security is in use the CICS DFLTUSER (for a CICS Transaction Server for OS/390 version 1.3 or later system) or the CWBM transaction userid (for a pre-CICS Transaction Server for OS/390 version 1.3 system) must be given access to the COVP, COVU, and COVE transactions.

**Security access for the administrator:** The userid that starts the Web User Interface (terminal user of COVC or PLTPIUSR, if started automatically via PLTPI) must have access to the COVC and COVG transactions.

If CICS surrogate user security checking is active in the Web User Interface server region, the userid that started the Web User Interface (terminal user of COVC or PLTPIUSR, if started automatically via PLTPI) must have READ access to wui-userid.DFHSTART in the SURROGAT class for all Web User Interface users.

**Security access for the end-user and users of the View Editor:** The Web User Interface end-user needs access to the COVA transaction and CICSplex SM.

Users of the View Editor need access to the COVA transaction, CICSplex SM and the View Editor profile. For more information about access to the View Editor, see the *CICSplex System Manager Web User Interface Guide*.

All users who are successfully signed onto the Web User Interface have access to all of the customizable view and menu help pages, if the customizable view and menu help is served by the Web User Interface.

**Summary:**

Table 20 summarizes the security accesses required by users of the Web User Interface.

*Table 20. Security accesses required by users of the Web User Interface*

User Roles	CICS Web Interface	Administrator	End-user	View Editor
Transactions	COVP COVE COVU	COVG COVC	COVA	COVA
CICS surrogate user security		Yes		
View Editor profile				Yes
CICSplex SM and CICS security			As appropriate for individual users	As appropriate for individual users

**Secure Sockets Layer support**

If you are using a CICS Transaction Server for OS/390 version 1.3 or later system, you can provide secure connections by using the Secure Sockets Layer (SSL) support to provide encryption on the connection. For information about SSL support, see the *CICS Internet Guide*. Also, see “Specify the Web User Interface server initialization parameters” on page 292 for information about the TCPIPSSL and TCPIPSSLCERT, Web User Interface server initialization parameters, that you need to specify for SSL support.

**Note:** Web User Interface SSL support uses server authentication only. User authentication is by the external security manager (ESM) user ID and password.

**Authorizing access to MVS data sets**

In addition to standard CICS and CICSplex SM requirements, the CICS region userid must have the authority to access the data sets associated with the DDnames described in Table 21.

*Table 21. Security access required for MVS data sets*

DDnames	Access required
---------	-----------------

Table 21. Security access required for MVS data sets (continued)

EYUWUI	READ
DFHHTML	READ
EYUCOVI (and clones)	READ
EYUWREP	UPDATE
EYULOG	UPDATE
EYUCOVE (and clones)	UPDATE

## Additional tasks

After you have set up the Web User Interface you need to:

### Submit start-up JCL

You should submit the start-up JCL for the Web User Interface server to start it for the first time. To confirm that the Web User Interface server has started successfully, you should ensure that the following messages are displayed on the job log:

```
08.52.33 JOB03331 +EYUVS0001I IYCQCTA5 CICSPlex SM Web User Interface
                                initialization started.
08.52.37 JOB03331 +EYUVS0002I IYCQCTA5 CICSPlex SM Web User Interface
                                initialization complete.
```

If you chose not to start the Web User Interface server during PLTPI processing, you can start it using the COVC transaction Start command.

### Check browser connection

You should check the connection between the Web User Interface and the web browser by typing in the following URL:

```
http://hostname:port/CICSPlexSM
```

where *hostname* is the name specified on the TCPIPHOSTNAME Web User Interface server initialization parameter and *port* is the value specified on the TCPIPPORT Web User Interface server initialization parameter. For information about the Web User Interface server initialization parameters see “Specify the Web User Interface server initialization parameters” on page 292. You should be presented with the Web User Interface server Welcome panel containing a Begin Signon button.

### Obtain view and menu definitions

Once the Web User Interface has been started, you should obtain some view and menu definitions. You can do this by either:

- Logging onto the CICS terminal and running the COVC transaction. The Import option of the COVC transaction enables you to import the starter set views and menus. The starter set views and menus are provided in CICSTS32.CPSM.SEYUVIEW. For information about the COVC transaction, see the *CICSPlex System Manager Web User Interface Guide*.
- Using the View Editor. For information about the View Editor see the *CICSPlex System Manager Web User Interface Guide*.

### Shutdown the Web User Interface server

You can shutdown the Web User Interface server by:

- Shutting down the CICS system in which the Web User Interface server is running.

- Using the COVC transaction. For more information about the COVC transaction see the *CICSplex System Manager Web User Interface Guide*.

---

## CICSplex SM system parameters

This chapter describes the system parameters that you can use to identify or alter CICSplex SM attributes.

These parameters are specified by means of an extrapartition transient data queue. The transient data queue name is COPR. The parameters may be assigned to a DD \* file, sequential data set or a partitioned data set member. The DD name for the extrapartition transient data queue is EYUPARM.

The system parameters are coded as 80-byte records. Multiple system parameters may be specified on a single record as long as they are separated by commas and do not exceed a total of 71 characters in length. The format of the system parameters is:

keyword(v)

where:

**keyword**

is the name of a CICSplex SM system parameter.

**v** is an alphanumeric data value that may be specified with the system parameter.

Table 22 identifies the CICSplex SM parameters used in the CMAS and MAS and indicates whether these parameters are required or optional.

For CMASs, MASs, and WUI servers, members of the TDFHINST and XDFHINST libraries containing samples of these parameters are as follows:

**EYUCMS0P**

CMAS parameters

**EYULMS0P**

Local MAS parameters

EYUCMS0P and EYULMS0P are supplied in TDFHINST (uncustomized) and XDFHINST (customized).

**Note:** Before using these members to start a CMAS, MAS, or WUI server remove the comments from the samples and supply the appropriate values.

*Table 22. CICSplex SM parameters used in CMAS, MAS, and WUI server*

Parameter	CMAS	MAS and WUI server	Default
ALERTRCVR	Optional	n/a	NETVALRT
ALERTVER	Optional	n/a	0
APISIGNMSG	Optional	n/a	YES
BASASSOCBLK	Optional	n/a	14301
BASLOGMSG	n/a	Optional	NO
CICSplex	n/a	Required	
CMASYSID	n/a	Optional	
CMTMLNKACQ	Optional	n/a	RECONN
COHTTASKPRI	n/a	Optional	200

Table 22. CICSplex SM parameters used in CMAS, MAS, and WUI server (continued)

Parameter	CMAS	MAS and WUI server	Default
COIRTASKPRI	n/a	Optional	200
COMMTSBLOCKS	Optional	Optional	128 (MAS) 512 (CMAS)
HISTORYONLY	n/a	Optional	NO
HISTRECSMSG	n/a	Optional	0
HISTSECS	n/a	Optional	30
JRNLDEFCH	Optional	n/a	NO
JRNLOPACT	Optional	n/a	NO
JRNLRTAEV	Optional	n/a	NO
MASALRLRTCNT	n/a	Optional	0
MASALRLRTPRI	n/a	Optional	255
MASALRLRTTIM	n/a	Optional	10
MASINITTIME	n/a	Optional	10
MASPLTWAIT	n/a	Optional	NO
MAXAUXCPSM	Optional	n/a	50
MAXAUXTOTL	Optional	n/a	70
MAXHISTRECS	n/a	Optional	1
MSGBUCKETS	n/a	Optional	1024
MSGCASE	Optional	Optional	Mixed
NAME	Optional	Optional	
RESSTATUS	Optional	n/a	NOTIFY
SEC	Optional	n/a	NO
SECLOGMSG	Optional	n/a	NO
SECTIMEOUT	Optional	n/a	30
SPOOLCLASS	Optional	Optional	P
STALLDBCTSK	n/a	Optional	4
STALLDBCCNT	n/a	Optional	4
STALLDB2TSK	n/a	Optional	4
STALLDB2CNT	n/a	Optional	4
STALLDLITSK	n/a	Optional	4
STALLDLICNT	n/a	Optional	4
STALLDSPTSK	n/a	Optional	4
STALLDSPCNT	n/a	Optional	2
STALLEJBCNT	n/a	Optional	3
STALLEJBTSK	n/a	Optional	2
STALLENQTSK	n/a	Optional	4
STALLENQCNT	n/a	Optional	4
STALLFLETSK	n/a	Optional	3
STALLFLECNT	n/a	Optional	4
STALLILKTSK	n/a	Optional	2
STALLILKCNT	n/a	Optional	3
STALLLCKTSK	n/a	Optional	4
STALLLCKCNT	n/a	Optional	4
STALLLGRCNT	n/a	Optional	3
STALLLGRTSK	n/a	Optional	2
STALLITVTSK	n/a	Optional	0
STALLITVCNT	n/a	Optional	0
STALLJNLTSK	n/a	Optional	1
STALLJNLCNT	n/a	Optional	3
STALLMQSCNT	n/a	Optional	3
STALLMQSTSK	n/a	Optional	2
STALLPGMTSK	n/a	Optional	2



Table 22. CICSplex SM parameters used in CMAS, MAS, and WUI server (continued)

Parameter	CMAS	MAS and WUI server	Default
STALLPGMCNT	n/a	Optional	4
STALLSESTSK	n/a	Optional	2
STALLSESCNT	n/a	Optional	3
STALLSTGTSK	n/a	Optional	1
STALLSTGCNT	n/a	Optional	2
STALLTDQTSK	n/a	Optional	3
STALLTDQCNT	n/a	Optional	4
STALLTRMTSK	n/a	Optional	0
STALLTRMCNT	n/a	Optional	0
STALLTSKTSK	n/a	Optional	0
STALLTSKCNT	n/a	Optional	0
STALLTSQTSK	n/a	Optional	3
STALLTSQCNT	n/a	Optional	4
STALLUSRCNT	n/a	Optional	0
STALLUSRYSK	n/a	Optional	0
STALLWEBCNT	n/a	Optional	3
STALLWEBTSK	n/a	Optional	2
STALLXMGTSK	n/a	Optional	4
STALLXMGCNT	n/a	Optional	2
STALLXRFTSK	n/a	Optional	1
STALLXRFCNT	n/a	Optional	2
SUPPRESSCMF	n/a	Optional	NO
TOBATCHREQ	Optional	n/a	0
TOONLINEREQ	Optional	n/a	0
TOPOLLINT	Optional	n/a	300
WLMLOADCOUNT	n/a	Optional	ALLQUEUED
WLMLOADTHRSH	n/a	Optional	65

**ALERTRCVR(NETVALRT | name)**

Identifies the 1-8 character name of the NetView Alert Receiver to be used by the CMAS if NetView Generic Alerts are to be sent by the CMAS to NetView.

**ALERTVER( 0 | 1 )**

Identifies the version of the CPSM generic alert record that the CMAS will send to NetView. See Appendix B of *CICSplex System Manager Managing Resource Usage* manual for details on the generic alert Records CPSM may send to NetView.

ALERTVER is only relevant for a CMAS which is named in an ACTNDEF as sending generic alerts to NetView.

**APISIGNMSG(YES | NO)**

Indicate whether the successful signon / signoff message, EYUXD0807I, is to be issued when a CICSplex SM API user CONNECTs to, or DISCONNECTs (TERMINATEs) from, the CICSplex SM API.

**BASASSOCBLK(value | 14301)**

Specify the number of BAS association blocks that can be acquired from a single association element. The default EYUPARM value creates an association segment size requiring approximately 1.2 mB of storage. If the maximum value of 114597 blocks-per-segment is specified, then the resultant segment size will be just over 8 mB.

**BASLOGMSG(YES | NO)**

Indicate whether CICS resources defined via BAS should have their definitions logged to the CSDL Transient Data Queue of the MAS when they are installed.

If the CICS version used by the MAS does not support the LOGMESSAGE option of the EXEC CICS CREATE command, BASLOGMSG will have no effect.

**CICSPLEX(name)**

Identify the 1- to 8-character name of the CICSplex to which the local MAS is to be associated.

The name of a CICSplex should not be the same as the name of a CMAS, a CICS system, or a CICS system group.

**CMASYSID(name)**

Identify the 1- to 4-character name of the CMAS to which a MAS is to be attached.

You may also use this parameter when a local MAS is to attach to a specific CMAS in the same MVS image.

**CMTCMLNKACQ(ALWAYS | RECONN)**

Specify whether you want the CMAS to attempt to reacquire LU6.2 CMAS to CMAS links (CMTCMLNKs) if the initial acquire attempt made by CICS fails. The initial acquire attempt is made by CICS when a CMAS is started, a VTAM ACB is opened, or a CMAS to CMAS definition (CMTCMDEF) is installed. If the initial acquire attempt fails, CICSplex SM network surveillance might attempt to re-acquire the LU6.2 CMAS to CMAS links depending on the value of CMTCMLNKACQ:

- If CMTCMLNKACQ is set to ALWAYS, the CMAS attempts to acquire CMTCMLNKs, independent of whether the links were established in the current CMAS run.
- If CMTCMLNKACQ is set to RECONN, the CMAS attempts to reacquire CMTCMLNKs only where CMTCMLNKs were established in the current CMAS run.

If the CMTCMLNKs were not acquired in the current CMAS run and the initial acquire attempts fail, CMTCMLNK must manually be acquired.

Depending on the CMTCMLNKACQ option that you are using, repeated failed acquire attempts can produce messages DFHZC3437, DFHZC3462 and DFHZC2405 and VTAM IST663 and IST664.

CMTCMLNKACQ can be dynamically changed in a CMAS using the COD0 SET command.

**COHTTASKPRI( value | 200)**

Specify the CICS task priority for the MAS COHT task. COHT is invoked in a MAS when an API or web user interface query for completed task history records (HTASK records) is directed to the MAS. Use this parameter to tune the priority of HTASK requests so that a resource intensive query does not impact the performance of other tasks in the MAS.

**COIRTASKPRI(value | 200)**

Specify the task priority of COIR, in the range 0 to 255. COIR is a CICSplex SM task that can be used to process evaluation definitions (EVALDEFs) independent of the MAS.

For each EVALDEF that requests a separate task, an instance of COIR is started at the specified priority. If you specify a priority of 0, no separate COIR tasks are started; all EVALDEFs are processed by the MAS long running task (LRT).

**Note:** This parameter does not apply to CICS for Windows.

### **COMMTSBLOCKS(value | 512/128)**

This is the number of sets of control blocks allocated at CMAS or MAS startup for CPSM Communications Transport Services. These control blocks are used when data must be shipped between a CMAS or MAS and other CMASs or MASs.

The default and minimum values for this parm are 512 in a CMAS and 128 in a MAS. The maximum value is 8192 in either a CMAS or MAS.

Each set requires 1204 bytes of storage allocated in ESDSA in the CMAS or MAS. If the defaults are used, the following total storage is allocated:

CMAS -  $512 * 1204 = 616,448$  bytes  
MAS -  $128 * 1204 = 154,112$  bytes

If the maximum value is specified, the following total storage is allocated:

CMAS -  $8192 * 1204 = 9,863,168$  bytes  
MAS -  $8192 * 1204 = 9,863,168$  bytes

If a shortage occurs during the execution of a CMAS or MAS, message EYUCT0105E is issued. At termination of the CMAS or MAS, message EYUCT0106W is issued. The later message will include a value equal to the highest concurrent shortage of sets (High water mark). It is recommended that the COMMTSBLOCKS parm for the CMAS or MAS be increased by at least the amount specified by the EYUCT0106W message before restarting the CMAS or MAS.

### **HISTORYONLY( YES | NO)**

Specify whether history data should be collected without collecting normal CICSplex SM monitoring data as well. For example, if MLOCTRAN and MREMTRAN data is not required set HISTORYONLY(YES) to prevent this data from being collected.

### **HISTRECSMSG( value | 0)**

Specify that message EYUNL0179I 'Task History Recorder dataset EYUHISTx has accrued nnnn records' should be output each time 'value' thousand records are written to the history data sets. The maximum allowed value is 1000 which means output a message every time 1 million records are written to the history data sets.

The message can be used as an aid to determining the optimum size of the history data sets. A value of 0 means no EYUNL0179I messages are to be produced.

### **HISTSECS( value | 30)**

Specify the number of seconds to use as the default when API or web user interface users specify a parameter of RECENT(HISTSECS) when requesting completed task (HTASK) resource table records. The maximum allowed value is 86400 seconds (24 hours).

### **JRNLDEFCH(YES | NO)**

Causes a journal record to be written for each data repository add, delete, and update operation.

**JRNLOPACT(YES | NO)**

Causes a journal record to be written for each successful action command issued against a MAS or CMAS.

**JRNLRRTAEV(YES | NO)**

Causes a journal record to be written each time an real-time analysis (RTA) event is generated.

**MASALTLRTCNT(0 - 5 | 0)**

The number of alternate long running tasks (CONA) started in the MAS during MAS agent initialization. These tasks remain active until the MAS agent terminates or goes into restart mode, and handles all API/WUI/RTA normally handled by the CONL task, allowing the CONL task to perform other processing in the MAS. At any time, only one of the CONA tasks processes requests. If the CONA task that is currently processing requests becomes busy (as determined by the value of the MASALTLRTTIM EYUPARM), subsequent requests are directed to another CONA task.

If zero (0) is specified, no CONA tasks are started and the CONL task services the API/WUI/RTA requests that are normally directed to the long running task.

**Note:** Specifying different values for MASALTLRTCNT for multiple WLM target regions might result in an uneven distribution of transactions to those regions because of differing long running task counts.

**MASALTLRTPRI(0 - 255 | 255)**

The priority given to the CONA transaction for the current execution of the MAS.

**Note:** Specifying this value less than 255 can adversely affect the response time of API, and WUI users, and might result in RTA EVENTS not being created or resolved in a timely manner

**MASALTLRTTIM(1 - 3600 | 10)**

The amount of time in seconds for which a CONA task can be busy before subsequent requests are directed to another active CONA task.

**MASINITTIME(value | 10)**

Specifies the number of minutes, from 5 to 59, that CICSplex SM should wait for the MAS to initialize.

- If you specify MASPLTWAIT(YES), the MASINITTIME value is the maximum length of time that PLT processing can be suspended for MAS initialization. (By suspending PLT processing the chance of completing MAS initialization within a specified time is increased, because you are asking for less work to be done in a given time interval, and reducing the scope for contention, during that time).
- If you specify MASPLTWAIT(NO), the MASINITTIME value is the maximum length of time that can elapse before MAS initialization is halted if it does not complete.

**MASPLTWAIT(YES | NO)**

Indicates whether CICSplex SM should suspend all PLT processing until the MAS is fully initialized and connected to the CMAS.

- When you specify MASPLTWAIT(YES), no CICS applications can be started and no users can sign on to the system until CICSplex SM completes the installation of resources and resumes PLT processing. If CICSplex SM does complete the installation of resources and resume PLT processing within the time interval specified by MASINITTIME,

message EYUTS0003I is issued. If it does not complete within the time interval specified by MASINITTIME, message EYUNL0090W is issued, the MAS initialization is halted and the PLT processing resumes to allow the region to function as a CICS region without CICSplex SM control. MAS Initialization can be retried by entering the COLM transaction manually.

- When you specify MASPLTWAIT(NO), CICSplex SM still observes the MASINITTIME value waiting for the MAS agent to complete the topology connect. If CICSplex SM does not complete the topology connect within the time interval specified by MASINITTIME (or its default value), message EYUNL0090W is issued, the MAS initialization is halted and the PLT processing resumes to allow the region to function as a CICS region without CICSplex SM control. MAS Initialization can be retried by entering the COLM transaction manually.

If you are using Business Application Services (BAS) to automatically install resources at CICS system initialization, you should specify MASPLTWAIT(YES) for that system.

**Note:** If you are using Business Application Services (BAS) to automatically install a DB2 connection, and you want the connection to be activated during CICS startup, see the information on page “Activating DB2 connections during CICS startup” on page 279

#### **MAXAUXCPSM(value | 50)**

Specify the percent of total auxiliary storage which may be committed to each CMAS, in the range of 0 to 99. A value of 0 stops CPSM from checking the amount of space used.

**Important:** Setting a value of 0 might result in a shortage of auxiliary storage, requiring an IPL of the MVS system.

Each CMAS requires 24,160 4 KB pages (94 MB) of cache storage at initialization. If a request for additional cache storage would cause the CMAS to exceed the MAXAUXCPSM threshold, an SDUMP is taken and the CMAS is terminated. If this occurs during CMAS initialization, it means that the CMAS was unable to acquire the initial allocations for all required component data cache areas. Either the value of MAXAUXCPSM must be increased, or the total amount of auxiliary storage must be increased by adding or expanding external page data sets. If this threshold is reached during an attempt to create or extend a data cache after CMAS initialization has completed, ARM is invoked to attempt to restart the CMAS.

#### **MAXAUXTOTL(value | 70)**

This value is the maximum total auxiliary storage usage at which the CMAS will allow a request for additional cache storage to be made, in the range of 0 to 99. A value of 0 stops CPSM from checking the amount of space used.

**Important:** Setting a value of 0 might result in a shortage of auxiliary storage, requiring an IPL of the MVS system.

Setting a nonzero value for MAXAUXTOTL prevents the CMAS from requesting an amount of cache storage that would cause the MVS system to enter a state of auxiliary storage shortage. If a request for additional cache storage would cause the CMAS to exceed this threshold, an SDUMP is taken and the CMAS is terminated. This parameter may cause a CMAS to shut down even though the CMAS is not the largest user of auxiliary storage. If this occurs during CMAS initialization, it means that the CMAS

was unable to acquire the initial allocations for all required component data cache areas. The total amount of auxiliary storage available must be increased by adding or expanding external page data sets. If this threshold is reached during an attempt to create or extend a data cache after CMAS initialization has completed, ARM is invoked to attempt to restart the CMAS.

**MAXHISTRECS( value | 1)**

Specify a value in thousands, in the range 1 to 50. This is a limit on the number of records returned on a completed task query from the MAS. This can be used to limit the amount of data in a request for completed task (HTASK) resource table records. When this limit is reached, the CICSplex SM API GET request will receive a WARNING response and MAXRECORDS reason.

**MSGBUCKETS(value | 1024)**

Specify a value in the range 1 to 32768. This value specifies the number of buffers to be allocated for Topology data collection in the MAS. Each buffer is 64 bytes long. The buffer pool, allocated in the MAS cache data space, is used by CPSM's XMEOUT and XRSINDI Global User Exits and by the MAS Heartbeat task. The number of buffers must equal or exceed the total number of connections, DSNAMEs, GLUEs, TRUEs, and FEPI connections defined in the MAS. If the number of buffers is not sufficient for the Topology mapped resources in the MAS, a trace record with debug text XDATLOST is written at every other heartbeat interval when MAS Topology resource data is collected. If this occurs, resources will be missing from Topology Resource Maps for the MAS in all CMASes in the CICSplex, and query or action requests entered from the WUI or API for specific resources may fail because the target resources are not known to Topology.

**MSGCASE(MIXED | UPPER)**

Indicate whether the following types of output should be displayed in mixed case or upper case:

- Messages issued by Message Services to the console, joblog, and EYULOG
- Batched repository-update facility output
- Diagnostic output from the CODB, COD0, and COLU transactions.

You can specify:

**MIXED**

Mixed case text is displayed as is. If you specify MIXED, output may be displayed incorrectly on Katakana display terminals, where lower case characters are displayed as Katakana symbols.

**UPPER**

Mixed case text is displayed in upper case only.

**NAME(name)**

Identify the 1- to 8-character name of the CMAS or local MAS that is to be started. If you do not specify this parameter, the default is the VTAM application ID.

**RESSTATUS(NOTIFY | MSG | CONMSG)**

Indicate how the CMAS is to respond when a CICS resource that is being reported to the resource status facility has a change in operational state:

**NOTIFY**

Issues event notifications in the form of ERESSTAT resource table records.

These event notifications can be monitored by using the LISTEN command of the CICSplex SM API. For more information, see *CICSplex System Manager Application Programming Guide*.

**MSG** Writes external messages to EYULOG.

If you specify MSG, event notifications are produced in addition to the messages.

**CONMSG**

Writes external messages to the job log, console, and EYULOG.

If you specify CONMSG, event notifications are produced in addition to the messages.

**Note:** Use this option with care. It could cause a large number of messages to be sent to the console.

**SEC(YES | NO)**

For a CMAS, indicate whether the CMAS is to perform security checking of CICSplex SM requests directed to the CICS systems it manages.

When NO all security-related parameters are ignored.

If a CMAS manages any CICS regions that are running with security active (SEC=YES specified as a system initialization parameter), the CMAS must include SEC(YES) in EYUPARM. If you do not activate CICSplex SM security in the CMAS, a connection cannot be established to a CICS system that specifies SEC=YES. If a connection is attempted, the following message is issued to the console, the CMAS job log, and the CMAS EYULOG:

```
EYUCR0007E Security mismatch between CMAS cmasname and
           MAS masname. Connection terminating.
```

**Note:** If a CMAS started with SEC(NO) connects directly or indirectly to a CMAS started with SEC(YES), any request sent to the SEC(YES) CMAS will fail.

- If the request originates from the CICSplex SM API connected to the SEC(NO) CMAS, the API request will receive: RESPONSE 1031 NOTPERMIT REASON 1345 USRID
- If the request originates from the CICSplex SM Web User Interface server connected to a SEC(NO) CMAS, the browser will receive message: EYUVC1220E

**SECLOGMSG(NO | YES | ALL)**

Controls whether CICSplex SM should issue message EYUCR0009I to the CMAS EYULOG, to record security failures.

When NO is specified, the default, message EYUCR0009I is not issued.

Specify YES, or ALL, to cause message EYUCR0009I to be issued.

SECLOGMSG(YES) can be useful if the External Security Manager (ESM) does not issue messages when it cannot make a decision, or when a failure occurs.

**Note:** When SECLOGMSG(YES) is specified, EYUCR0009I will be issued only for requests which are to be logged to the ESM.

SECLOGMSG(ALL) causes EYUCR0009I to be issued even when the ESM permits access to the resource. The ALL operand can produce a large number of EYUCR0009I messages and should normally be used only under the direction of IBM Support.

SECLOGMSG can be changed dynamically in a CMAS with the COD0 SET command.

### **SECTIMEOUT(value|30)**

Specifies the time in minutes, in the range of 1 through 1440 (1 day), that idle user IDs are to remain signed on within the CMAS before being considered for timeout.

The value is also used to control how often the CMAS checks for idle users to timeout. For example, with the default value of 30, the CMAS checks every 30 minutes for users who have not used the CMAS for 30 minutes. However, as the times are not synchronized it may mean that the user ID is not timed out for up to double the SECTIMEOUT value. Setting this value low increases the number of calls to the External Security Manager (ESM). Setting this value high means that users may have to wait a long time before automatically picking up security changes that affect the user ID (for example, adding the user to a new group).

The CMAS or CMASLIST PURGE request (available from the API and WUI) can be used to force a CMAS to check for users to timeout immediately.

The CMAS or CMASLIST RESET USERID request (available from the API and WUI) can be used to force the CMAS to rebuild the user's security information the next time it is used. This request would typically be used after adding or removing a user ID to or from a group, and the user does not want to wait to be timed out to pick up the change.

### **SPOOLCLASS(class | P)**

Specify a SYSOUT class value, from A to Z, that identifies where CICSplex SM spool output is to be sent.

Spool output can be generated by these CICSplex SM functions:

- The online utility transaction (COLU)
- The PRINT and CAPTURE commands of the interactive debugging transaction (COD0).

### **STALLxxxTSK**

Where xxx represents a CICSplex SM suspend class. The values for xxx are shown in Table 23 on page 315.

Identify the minimum number of concurrent tasks required to enter the suspend class. The value may be between 0 and 999. Use 0 to indicate STALL detection for the xxx suspend class is not active. The default value for each task is shown in Table 22 on page 305.

### **STALLxxxCNT**

Where xxx represents a CICSplex SM suspend class. The values for xxx are shown in Table 23 on page 315.

Identify the number of consecutive occurrences of an entry in the suspend class required for CICSplex SM to report a STALL. The value may be between 0 and 999. Use 0 to indicate STALL detection for the xxx suspend class is not active. The default value for each task is shown in Table 22 on page 305.



Table 23. CICSplex SM Suspend Classes

Suspend Class	CICS Suspend Types	Value in STALLxxx Parameters	Text in EYUPNxxxx Messages
Allocate Session	ALLOCATE	SES	ALLCSESS
Console	CQSYSTEM	CON	CONSOLE
DBCTL	DBCTL	DBC	DBCTRL
DB2	CDB2RDYQ CDB2TCB DB2_INIT DB2 CDB2CONN DB2DISC DB2EDISA	DB2	DB2
DLI	DLI	DLI	DLI
Dispatcher	DS_HELD DISPATCH	DSP	DISP
Enterprise Java	EJ.ST.DJ. EJ.ST.DC.	EJB	EJB
Enterprise Java	RZRSTRAN	EJB	EJB
Enterprise Java	RZRSTRIG	EJB	EJB
Enterprise Java	SHREQEST	EJB	EJB
Enterprise Java	SHSYSTEM	EJB	EJB
Enqueue	KC_ENQ ENQUEUE	ENQ	ENQUEUE
File	FCxxxxxx CFDTxxxx	FLE	FILE
File	FCBFSUSP	FLE	FILE
File	FCWAITQ	FLE	FILE
Interval Control	ICxxxxxx	ITV	INTV
Interval Control	TIEXPIRY	ITV	INTV
IP Interconnectivity	IS_xxxx	IPC	IPIC
Journal	JASUBTAS JCxxxxxx	JNL	JOURNAL
Lock Manager	LMQUEUE	LCK	LOCK
Logger	LGxxxxxx	LGR	LOGGER
Log Manager	LGxxxxxx	LGR	LOGGER
MQSeries	MQSERIES	MQS	MQSERIES
Program Loader	APRDR	PGM	PROGRAM
Program Loader	CPI	PGM	PROGRAM
Program Loader	EDF	PGM	PROGRAM
Program Loader	PROGRAM	PGM	PROGRAM
Resource manager	DFHPTTW	RMI	RM

Table 23. CICSplex SM Suspend Classes (continued)

Suspend Class	CICS Suspend Types		Value in STALLxxx Parameters	Text in EYUPNxxxx Messages
Resource manager	PRM		RMI	RM
Resource manager	RMCLIENT		RMI	RM
Resource manager	RMUOWOBJ		RMI	RM
Resource manager	UNSHUNT		RMI	RM
Sockets domain	CCACHE		SOC	SOCKETS
Sockets domain	SOCKETS		SOC	SOCKETS
Storage	SMSYSTEM		STG	STORAGE
Storage	xDSA ExDSA		STG	STORAGE
Terminal	NOTI		ILK	IRLINK
Transaction manager	FOREVER		XMG	TRANSACT
Transaction manager	RESYNC		XMG	TRANSACT
Transaction manager	SOCBNOTI		XMG	TRANSACT
Transaction manager	SOCFNOTI		XMG	TRANSACT
Transaction manager	TRANDEF		XMG	TRANSACT
Transaction manager	XM CHILD		XMG	TRANSACT
Transaction manager	XM-PAREN		XMG	TRANSACT
Transient Data	MBCB_xxx TDEPLOCK TD_INIT	MRCB_xxx TDIPLOCK TD_READ	TDQ	TSDATA
Terminal Control	ZCxxxxxx		TRM	TERM
Task Wait	EKCWAIT KCCOMPAT		TSK	TASKWAIT
Temporary Storage	TSxxxxxx		TSQ	TEMPSTOR
Terminal	IRLINK		ILK	IRLINK
Terminal Control	AITM		TRM	TERM
Transaction Manager	XM_HELD TCLASS	MXT	XMG	TRANSACT
User wait	USERWAIT EDF		USR	USERWAIT
Web services	WBALIAS WEB_ECB		WEB	WEBSERV
Webservices	PIISLSTN		WEB	WEBSERV
Webservices	PIPELINE		WEB	WEBSERV
Webservices	RZCBNOTI		WEB	WEBSERV

Table 23. CICSplex SM Suspend Classes (continued)

Suspend Class	CICS Suspend Types	Value in STALLxxx Parameters	Text in EYUPNxxxx Messages
XRF	XRxxxxxx	XRF	XRF
<b>Note:</b> EYUPNxxxx messages are issued when a stall condition occurs that generates a real-time analysis system availability monitoring (SAM) event.			

### SUPPRESSCMF(YES | NO)

For a local MAS, indicates whether the records collected by the CICS Monitor Facility are written to SMF.

The parameter only suppresses CICS type 3 performance class records. Type 4 exception records and type 5 transaction resource records are not suppressed. The type 3 performance records are only suppressed if the CICS region has an active CICSplex SM monitor definition installed for the MTRAN monitoring class. You can verify which CICS regions have active monitoring for the MTRAN class using the WUI Active monitor specifications (POLMON) tabular view.

### TOBATCHREQ(value | 0)

The time in seconds before a batch request directed to a MAS is timed out. This includes RTA requests and API requests initiated from non-CICS programs. Specify zero, or a value in the range 10 to 1800.

- If you specify zero, the default value of 240 seconds (4 minutes) is applied. This value is then doubled when the request is transmitted to the MAS.
- If you specify a non-zero value in the range 10 to 1800, that value is used.
- If you specify a non-zero value less than 10, TOBATCHREQ is set to 10.

Depending upon the value specified for TOBATCHREQ more time outs may be received. This may be noticed in the following ways:

#### RTA requests

No data will be processed for any MAS that times out and no external message will be displayed. This may cause an event to not be created, or lead to premature termination of existing events.

#### API requests initiated from non-CICS programs

All API requests initiated from non-CICS programs will receive a REPSONSE of ENVIRONERROR (1030) and REASON of REQTIMEOUT (1342) and no data records will be returned, regardless of the CONTEXT and SCOPE of the request.

### TOONLINEREQ(value | 0)

The time in seconds before an online request directed to a MAS is timed out. This includes WUI requests and API requests initiated from CICS programs. Specify zero, or a value in the range 10 to 1800.

- If you specify zero, the default value of 240 seconds (4 minutes) is applied. This value is then doubled when the request is transmitted to the MAS.
- If you specify a non-zero value in the range 10 to 1800, that value is used.

- If you specify a non-zero value less than 10, TOONLINEREQ is set to 10.

Depending upon the value specified for TOONLINEREQ more time outs may be received. This may be noticed in the following ways:

#### **WUI requests**

All WUI requests will receive the following message and no data records will be displayed, regardless of the CONTEXT and SCOPE of the request.

#### **EYUVC1220E**

CICSplex SM API command (GET) failed. (Environerror, Revertimeout)

#### **API requests initiated from CICS programs**

All API requests initiated from CICS programs will receive a REPSONSE of ENVIRONERROR (1030) and REASON of REQTIMEOUT (1342) and no data records will be returned, regardless of the CONTEXT and SCOPE of the request.

#### **TOPOLLINT(value | 300)**

The time in seconds that a CMAS delays between checking all requests to determine if they exceed their time out time. Since polling is used to determine when to time out a request, more reliable time outs will occur if this value is set less than or equal to both TOBATCHREQ and TOONLINEREQ.

When a request is directed to a MAS that is not connected to the originating CMAS, it is transmitted from the originating CMAS to the remote CMAS to which the MAS is connected. When this occurs, the remote CMAS performs the time out processing, based upon the TOBATCHREQ and TOONLINEREQ values specified in the originating CMAS and the TOPOLLINT value specified in the remote CMAS. For this reason, time out processing will be more consistent if all CMASes in the network have the same TOPOLLINT value.

#### **WLMLOADCOUNT(ALLQUEUED | MXTQUEUED)**

For a local MAS, indicates which queued tasks should be included in the WLM task load count for the MAS.

Specify ALLQUEUED to include tasks queued for both maxtask and tranclass.

Specify MXTQUEUED to include tasks queued only for maxtask.

#### **WLMLOADTHRSH(65 | 1 - 100)**

For a local MAS that is defined as a CPSM WLM routing region, indicates what task load percentage ((active tasks/maxtask) \*100) should be used as part of the determination as to when the routing region should route to target regions on other CECs.

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

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

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



---

## Part 5. CICS verification

This part describes the processes and procedures you should follow to run the installation verification procedures for CICS.





---

## Chapter 32. CICS verification

This describes the installation verification procedures for CICS.

This part describes the processes and procedures you should follow to run the installation verification procedures for CICS. It contains the following chapters:

- “Running the installation verification procedures.”

---

### Running the installation verification procedures

After you have installed CICS, and applied any necessary service, you can use the CICS-supplied installation verification procedures (IVPs) to confirm that CICS is operational.

There are two IVP jobs:

#### 1. DFHIVPBT (verify batch)

This job starts up CICS, specifying a pair of sequential input and output devices (CARDIN and PRINTER) to be used instead of an ordinary terminal. It then executes a number of CICS transactions that are read from CARDIN. The last transaction in the input stream shuts down CICS.

#### 2. DFHIVPOL (verify online)

This job can run CICS with either XRF=NO, or XRF=YES. It is generated with XRF=NO specified as an override, which you change when you are ready to verify CICS with XRF.

You can use this CICS region to automatically install (**autoinstall**) an IBM 3270 Information Display System terminal, with which you can:

- Use the master terminal transaction, CEMT. You can also use CEMT from the MVS system console. For information about using CEMT, see the *CICS Supplied Transactions* manual.
- Use the resource definition online transaction, CEDA. For information about using CEDA, see the *CICS Resource Definition Guide*.
- Use the sample application transaction AMNU, to access the sample VSAM file, FILEA. For a description of the FILEA sample applications, see the *CICS 4.1 Sample Applications Guide*.

Before you run the IVP jobs, you will need to prepare your system. The steps you need to perform are described in “Preparation for running the IVPs.”

### Preparation for running the IVPs

Perform the following steps:

- Create the CICS data sets for the IVP jobs
- Install the CICS SVC for the IVP jobs
- Define and activate the CICS applids
- Authorize the IVP userid
- Review security requirements for the IVP jobs
- Define log streams
- Specify system initialization parameters for the IVP jobs

## Create the CICS data sets for the IVP jobs

Before you can run any of the CICS-supplied IVP jobs, create the data sets that they use. For further information about creating the data sets for the IVP jobs, see “Creating the CICS data sets” on page 218.

## Install the CICS SVC for the IVP jobs

All the IVP jobs require the CICS Type 3 SVC, which must be installed in the LPA. If you have not already installed the CICS SVC in the LPA (as described in Chapter 15, “Installing CICS modules in the MVS link pack area,” on page 129 ), do so now before attempting to run any of the IVP jobs. The IVP jobs do not use the Type 6 SVC (DFHHPSVC).

## Define and activate the CICS applids

If you want to use VTAM with a CICS region started by any of the CICS IVP jobs, create and activate a VTAM APPL definition for the CICS region's application identifier (applid). The applid defined to VTAM must match the applid that is specified on the APPLID system initialization parameter that is used by the IVP job. For example, to be able to logon to the CICS region that is started by the DFHIVPOL job, you must do one of the following:

- Create and activate an APPL definition for your own applid, which you specify on the APPLID parameter of the DFH\$SIP1 member of the SYSIN data set.
- Define and activate an APPL definition for the default applid DBDCCICS, which you specify on the APPLID parameter of the DFH\$SIP1 member of the SYSIN data set.

For more information about creating and activating VTAM APPL definitions for CICS, see Chapter 14, “Defining CICS regions as applications to VTAM,” on page 117 and “Naming conventions” on page 219.

Further, if you want to use VTAM cross-domain services to communicate between CICS regions on separate MVS images, you must create and activate VTAM CDRSC definitions in both MVS images involved in the communication. For more information about creating and activating VTAM CDRSC definitions for CICS, see “Cross-domain considerations” on page 119.

## Authorize the IVP userid

To run the IVP jobs with external security, you must define to RACF an IVP default CICS userid that has authority to run the transactions used as part of the IVP jobs. These transactions include the CICS-supplied transactions that are listed in Table 24 on page 325. The level of authority that is required by the IVP userid depends on what security you want to use for the IVP jobs. For more information, see “Review security requirements for the IVP jobs” on page 325.

**Note:** On a production system the default user should *not* have access to any CICS-supplied transactions except those you need in your CICS environment. The resource access authorizations that you give to the default user should clearly be limited to those resources that you intend to be universally available, and therefore not restricted in any way.

For information about the security requirements for CICS-supplied transactions, and about CICS security in general, see the *CICS RACF Security Guide*.

Table 24. Transactions used as part of the IVP jobs

Application	Transactions
DFH\$BTCH	CWTO, CEMT, CEOT, CSFE
FILEA samples	
DFH\$MNU	AMNU, MENU, PMNU, DMNU
DFH\$ALL	AINQ, INQY, PINQ, DINQ AADD, ADDS, PADD, DADD AUPD, UPDT, PUPD, DUPD
DFH\$xBRW	ABRW, BRWS, PBRW, DBRW
DFH\$xREN	AORD, OREN, PORD, DORD
DFH\$xCOM	AORQ, OREQ, PORQ, DORQ
DFH\$xREP	AREP, REPT, PREP, DREP
Other functions	CETR, CEDA, CMAC, CMSG, CSGM

## Review security requirements for the IVP jobs

You can run the IVP jobs with or without external security.

As supplied, the system initialization table, DFHSIT, used by the IVP jobs, specifies that external security is on. However, the IVP jobs have been set up with SEC=NO, indicating that external security is not on. DFHSIT also specifies that the IVP jobs are subject to transaction security (XTRAN=YES), resource security (Xyyy=YES), and command security (XCMD=YES).

**Note:** As supplied, the DFH\$SIP2 member of the SYSIN data set used by the DFHIVPBT job specifies the SIT override SEC=NO, so that you can run this job without external security.

If you choose to run the IVP jobs with external security, you must:

- Define CICS resource profiles to RACF.
- Define an IVP default CICS userid to RACF.
- Specify the IVP userid on the DFLTUSER=userid system initialization parameter.

You must also give the IVP userid sufficient authority to use transactions and resources that are needed to run the IVP jobs. That is, you must:

- Authorize the IVP userid to run the transactions that are used as part of the IVP jobs. (See Table 24.) To do this you must add the IVP userid, with READ access, to the access list of the RACF profiles for the transaction member class (TCICSTRN) or the transaction group class (GCICSTRN).

If you define the transactions as prefixed resources, you must also specify the system initialization parameter SECPRFX={YES |prefix} for the IVP jobs.

- Authorize the IVP userid to access the resources that are used by the transactions. To do this you must add the IVP userid, with appropriate authority, to the access list for the resource class profiles.
- Authorize the IVP userid to issue SP-type commands using the CEMT master terminal transaction. To do this, you must add the IVP userid, with appropriate authority, to the access list of the RACF profiles for the resource member class (CCICSCMD) or the resource group class (VCICSCMD). You must give the IVP userid UPDATE access for the SHUTDOWN resource class, otherwise the userid

will not be able to terminate the IVP jobs. You should also give the IVP userid UPDATE access for the DUMPDS and SYSTEM resource classes, if the DFHIVPBT job is to be run with external security.

For information about implementing external security, see the *CICS RACF Security Guide*. Alternatively, you can run the IVP jobs with limited security, for example:

- Without command security (XCMD=NO), the IVP userid would be able to run the IVP jobs without the need for authority to use the CEMT SP-type commands and the resources that they access.
- With only transaction security (Xyyy=NO including XCMD=NO), the IVP userid would be able to run the IVP jobs if authorized only to use the transactions used as part of the IVP jobs.

### Define log streams

CICS automatically attempts to connect to its system log stream, unless you define a journal model resource definition to define the log stream as TYPE(DUMMY). This means that you need to decide whether you want to run the IVPs with system logs, or to run with dummy logging.

If you decide to run with actual log streams, see Chapter 27, “Defining the logger environment for CICS journaling,” on page 169 for information about defining log streams.

Alternatively, you can define a CICS JOURNALMODEL resource definition with TYPE(DUMMY) to avoid having to define log streams. If you want to run the IVPs with the minimum effort:

- Define JOURNALMODEL resource definitions in the CSD for the primary and secondary system logs, DFHLOG and DFHSHUNT respectively, specifying TYPE(DUMMY); see Figure 27 on page 327 for a sample job.
- Add the CSD group that contains your dummy system log journal models to your own group list, and include your group list on the GRPLIST system initialization parameter.

Note that your group list must *follow* the IBM-supplied list DFHLIST. DFHLIST includes group DFHLGMOD (which contains DFHLOG and DFHSHUNT JOURNALMODEL definitions) Concatenating your list after DFHLIST ensures that your DUMMY definitions replace the IBM definitions.

```

//CSDLGSTR JOB 1,BELL,MSGCLASS=A,MSGLEVEL=(1,1),CLASS=A
//CSDUP EXEC PGM=DFHCSDUP,REGION=1M,PARM='CSD(READWRITE)'
//STEPLIB DD DSN=CICSTS32.SDFHLOAD,DISP=SHR
//DFHCSD DD DSN=CICSTS32.CICSH###.DFHCSD,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSABOUT DD SYSOUT=*
//SYSABEND DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSIN DD *
*
* DEFINE JOURNAL MODELS FOR CICS LOG STREAMS AS DUMMY
*
DEFINE JOURNALMODEL(DFHLOG) GROUP(LOGTEST)
DESCRIPTION(DEFINE SYSTEM LOG AS DUMMY)
 JOURNALNAME(DFHLOG)
 TYPE(DUMMY)
*
DEFINE JOURNALMODEL(DFHSHUNT) GROUP(LOGTEST)
DESCRIPTION(DEFINE SYSTEM LOG AS DUMMY)
 JOURNALNAME(DFHSHUNT)
 TYPE(DUMMY)
/*
//

```

Figure 27. Sample job to define DUMMY JOURNALMODELS for CICS system logs

### Specify system initialization parameters for the IVP jobs

All the IVP jobs use the system initialization parameters that are specified in the associated DFH\$SIPn member of the SYSIN data set. The DFH\$SIPn members, as supplied by CICS, default to the unsuffixed SIT, DFHSIT, and the resources defined to CICS are adequate only for a basic CICS region. For example, in the case of the DFHIVPOL job, the resources defined limit the number of terminals you can use.

The DFH\$SIPn members of the SYSIN data set also contain some system initialization parameters to exclude CICS resources not required by the IVP jobs, or to include some not specified by the default SIT.

One such parameter is TCT=5\$, specifying the CICS sample terminal control table, in the *hlq.SDFHLOAD* library. This TCT defines the pair of sequential input and output devices, CARDIN and PRINTER. (These are the only devices that are defined in DFHTCT5\$.)

The DFH\$SIPn members of the SYSIN data set may need to be edited for:

- The default SVC number is 216. To use a different SVC number, specify CICSVC=nnn in the appropriate DFH\$SIPn member. For more information about defining CICS SVCs, see “Defining the CICS SVCs to your MVS” on page 111.  
The IVP jobs do not require the Type 6 SVC.
- The applid used is CICSIVP1. To use a different applid, change the system initialization parameter (APPLID=CICSIVP1) in the appropriate DFH\$SIPn member.
- The IVP jobs had external security switched off. To run with security (SEC=YES), define a suitable default userid (for example IVPUSER) with the required authority to run the IVP transactions. Add DFLTUSER=IVPUSER in the appropriate DFH\$SIPn member. For more information about defining the IVP userid, see “Authorize the IVP userid” on page 324.

Transactions can be defined as prefixed resources by using the IVP userid, IVPUSER or any other prefix, as the prefix (for example, IVPUSER.CEMT or prefix.CEMT). To do this, add SECPRFX=YES, or SECPRFX=prefix in the appropriate DFH\$SIPn member for the IVP job.

This enables transactions to be run as part of the IVP jobs without affecting other CICS regions. For example, when the DFH\$BTCH batch stream is run, CICS sends authorization requests to RACF for the transactions and identifies them as IVPUSER.xxxx, where xxxx is the transaction ID (CWTO, CEMT, and so on).

- Language Environment support, for all the high-level language sample programs, was added as described in the *CICS System Definition Guide*. CICS requires either pre-defined CSD definitions, for the Language Environment modules, to be installed or Program autoinstall to be active.

The IVP jobs include the required DD statements for the Language Environment libraries as comments.

- The resources for the CICS messages facility were defined, as described in “Resources for the CICS messages facility, CMAC,” and the DFHCMAC resource group added to a group list used for the IVP jobs.
- The IVP jobs run with auxiliary trace switched on (AUXTR=ON), and the auxiliary trace data set switching facility set to switch once only (AUXTRSW=NEXT).

Other notes about changes to the system initialization parameters for the IVP jobs, and about the IVP jobs generally, are in the sections that describe the IVP jobs.

If you want to use system initialization parameters to modify or enhance the scope of the IVP jobs, see the *CICS System Definition Guide* for details about the parameters.

## Resources for the CICS messages facility, CMAC

You can use the CICS messages facility (CICS-supplied transaction CMAC) to provide the messages and codes descriptions online. Before you can use this facility, you must create and initialize the CICS messages data set DFHCMACD, define the resources needed by the facility, and make them available to your CICS region.

For information about creating and initializing the DFHCMACD data set, see “Creating the CICS messages data set, DFHCMACI job” on page 223.

The file DFHCMACD, managed by CICS file control, accesses the DFHCMACD data set. You must create a definition for this file in the CSD. The CICS-supplied definition for the DFHCMACD file and other resources that are needed by the CICS messages facility are in the CSD group DFHCMAC. The CICS startup procedure (in the IVP jobs) has a DD statement for the CMAC file, but for dynamic allocation you should copy the supplied resource definition for the DFHCMACD file and add the DSNNAME option.

You should specify the DFHCMAC group of resources for the CICS messages facility only in those CICS regions that need to use the facility; for example on some terminal-owning regions, but perhaps not on data-owning regions.

## The CICS startup procedure, DFHSTART

All the IVP jobs include a procedure to start up CICS. You can use this procedure as a basis for your own CICS startup procedures. This procedure, DFHSTART, comprises the following steps:

1. CICSNTL—determine whether CICS is to be started
2. DTCNTL—determine whether dump and trace analysis is to be performed
3. CICS—execute CICS
4. PRTDMPA—print any contents of the CICS DFHDMPA dump data set
5. PRTDMPB—print any contents of the CICS DFHDMPB dump data set
6. PRTAUXT—print any contents of the auxiliary trace DFHAUXT data set
7. PRTBUXT—print any contents of the auxiliary trace DFHBUXT data set.

The following symbolic parameters are defined in the IVP jobs:

**INDEX1**

is the high-level index of the CICS run-time data sets, as specified on the DSINFO parameter of the DFHISTAR job. **Default:** INDEX1=hlq

**INDEX2**

is the high-level index of the CICS load libraries, as specified on the INDEX parameter of the DFHISTAR job. **Default:** INDEX2=hlq

**REGNAM**

is the REGION name for a single or MRO region. **Default:** REGNAM=TR

**REG**

defines the MVS region size for the CICS step. **Default:** REG=32M

**START**

is the type of CICS startup to be performed. **Default:** START=AUTO

**DUMPTR**

specifies whether dump and trace analysis is required.  
**Default:** DUMPTR=YES

**RUNCICS**

specifies whether CICS is to be started. **Default:** RUNCICS=YES

**OUTC**

is the output print class. **Default:** OUTC='\*'

**SIP**

is the suffix of the DFH\$SIP member (in the SYSIN data set) to be used during CICS startup. **Default:** SIP=T

**Note:**

1. The step CICS (to start up CICS) is executed only if you code RUNCICS=YES (the default). Code RUNCICS=NO if you want to perform dump and trace analysis without starting CICS.
2. The steps PRTDMPA, PRTDMPB, DFHAUXT, and DFHBUXT are executed only if you specify DUMPTR=YES (the default).
3. When you run the DFHISTAR job, it overrides the default values in the IVP jobs with the values you specified in the DFHISTAR job.

**DD statements for CICS data sets**

The startup job step contains DD statements for the CICS data sets that are listed in Table 25.

*Table 25. DD statements for CICS data sets in the DFHSTART procedure*

DDname	Description
SYSIN	SYSIN data set, containing the DFH\$SIPn members that specify system initialization parameter overrides.
DFHMACD	Messages data set, needed for the CICS messages transaction, CMAC.

Table 25. DD statements for CICS data sets in the DFHSTART procedure (continued)

DDname	Description
FILEA	Sample VSAM data set, needed by the FILEA sample applications.
DFHTEMP	Auxiliary temporary storage data set, needed by the FILEA sample applications.
DFHINTRA	Transient data intrapartition data set, needed by the FILEA sample applications.
DFHAUXT DFHBUXT	First auxiliary trace (A) data set. Second auxiliary trace (B) data set.  The auxiliary trace data sets, DFHAUXT and DFHBUXT, are needed because the IVP jobs run with auxiliary trace switched on, and the auxiliary trace data set switching facility set to switch once only.
DFHLCD	(Mandatory) CICS local catalog data set (VSAM), used by the CICS domains to save some of their information between CICS runs, and to preserve this information across a cold start.
DFHGCD	(Mandatory) CICS global catalog data set (VSAM), has a variety of uses, including: during a CICS run, holding resource definitions that are installed; and, during a controlled shutdown, recording part of the warm keypoint information.
DFHCXRF	Transient data extrapartition data set, used by CICS as the target for messages sent to any transient data destination before CICS has completed intrapartition transient data initialization. Use of this DDname is optional, but if it is not used, any messages that would have been written here are lost.
DFHLRQ	The local request queue data set is used to store pending BTS requests; for example, timer requests or requests to run activities. It is recoverable and used to ensure that, if CICS fails, no pending requests are lost. For more information, see the <i>CICS Business Transaction Services</i> .
LOGUSR	Data set for the extrapartition transient data destination, LOGA, used by the CICS sample programs.
MSGUSR	Data set for the extrapartition transient data destination, CSSL, used by a number of CICS services.
PLIMSG	Data set for the extrapartition transient data destinations used by PL/I application programs. This data set is the destination for PL/I statistics and messages (CPLI).
COUT	Data set for the extrapartition transient data destinations used by C/370 application programs. This data set is the destination for the C/370 output data streams, stdout (CCSO) and, indirectly, stderr (CCSE).
DFHDMPA DFHDMPB	First transaction dump (A) data set. Second transaction dump (B) data set. The dump data sets are included because CICS always tries to open a transaction dump data set, and issues a warning message if it is unable to do so for any reason.
DFHCSD	(Mandatory) CICS system definition data set (VSAM).

## Verify batch job, DFHIVPBT

The CICS-supplied verify batch job, DFHIVPBT, is tailored to your CICS environment and stored in the *hlq.XDFHINST* library when you run the DFHISTAR job.

This IVP comprises the following job steps:



1. **Job step GENINPT** unloads the member DFH\$BTCH from the *hlq.SDFHSAMP* library into the CARDIN data set (using the MVS utility program, IEBGENER).
2. **Job step DFHSTART** invokes the CICS initialization program, DFHSIP, to startup CICS. The DFHSIP program reads startup system initialization parameters from the DFH\$SIP2 member of the SYSIN data set.

The DFH\$BTCH data set (see Figure 28) is used as terminal input, and this should produce a printout similar to the sample output shown in Figure 30 on page 334.

```

CWTO START OF BATCH STREAM DFH$BTCH\
CEMT S TIME(120)\           EXIT TIME INTVL TO 120 MILLISEC
CEMT S DUMPDS SWITCH\      SWITCH FROM DUMP A TO B
CEOT\                       INQUIRE TERMINAL STATUS
CSFE\                       F. E. TERMINAL TEST REQUEST
PRINT\                      TO SEND ALL CHARACTERS
THIS MESSAGE HAS BEEN RECEIVED FROM THE TERMINAL AND IS BEING SENT BACK\
END\                         TO END THE TEST
CSXX\                       INVALID TRANSACTION IDENT.
CWTO END OF BATCH STREAM DFH$BTCH - SHUTTING DOWN CICS\
CEMT P SHUT\                NORMAL TERMINATION OF CICS

```

where \ is the End Of Data Input character X'E0'.

Figure 28. DFH\$BTCH data set, input to the DFHIVPBT job

### Sample job log for the DFHIVPBT job

When you run the DFHIVPBT job, your job log should look similar to the example shown in Figure 29 on page 332.

1  
0

J E S 2 J O B L O G -- S Y S T E M M V 2 6 -- N O D E W I N M V S 2 C

16.24.15 JOB35409 ---- TUESDAY, 24 APR 2007 ----  
16.24.15 JOB35409 ICH70001I CICINST LAST ACCESS AT 16:23:52 ON TUESDAY, APRIL 24, 2007  
16.24.15 JOB35409 \$HASP373 DFHIVPBT STARTED - INIT 60 - CLASS A - SYS MV26  
16.24.15 JOB35409 IEF403I DFHIVPBT - STARTED - TIME=16.24.15  
16.24.15 JOB35409 - --TIMINGS (MINS.)-- ----PAGING COUNTS----  
16.24.15 JOB35409 -JOBNAME STEPNAM PROCSTEP RC EXCP CPU SRB CLOCK SERV PG PAGE SWAP VIO SWAPS STE  
16.24.15 JOB35409 -DFHIVPBT GENINPT 00 28 .00 .00 .00 175 0 0 0 0 0  
16.24.15 JOB35409 -DFHIVPBT CICS CICS\_CNTL 01 20 .00 .00 .00 168 0 0 0 0 0  
16.24.15 JOB35409 -DFHIVPBT CICS DTCNTL 01 19 .00 .00 .00 154 0 0 0 0 0  
16.24.15 JOB35409 DFHPA1101 CICSIVP1 DFHSIT IS BEING LOADED.  
16.24.15 JOB35409 DFHPA1108 CICSIVP1 DFHSIT HAS BEEN LOADED. (GENERATED AT: MM/DD= 01/12 HH:MM= 13:57).  
16.24.15 JOB35409 DFHPA1100 CICSIVP1 OVERRIDE PARAMETERS FROM JCL EXEC STATEMENT: START=AUTO,SYSLN  
16.24.15 JOB35409 DFHPA1102 CICSIVP1 OVERRIDE PARAMETERS FROM SYSIN: 1  
16.24.15 JOB35409 DFHPA1927 CICSIVP1 TCT=5\$, SEQUENTIAL DEVICES 00020000  
16.24.15 JOB35409 DFHPA1927 CICSIVP1 TS=(,0), 00030000  
16.24.15 JOB35409 DFHPA1927 CICSIVP1 AUXTR=ON, 00040000  
16.24.15 JOB35409 DFHPA1927 CICSIVP1 AUXTRSW=NEXT, 00050000  
16.24.15 JOB35409 DFHPA1927 CICSIVP1 FCT=NO, 00070000  
16.24.15 JOB35409 DFHPA1927 CICSIVP1 XRF=NO, 00090000  
16.24.15 JOB35409 DFHPA1927 CICSIVP1 SEC=NO, 40000000  
16.24.15 JOB35409 DFHPA1927 CICSIVP1 SRT=NO, 40000100  
16.24.15 JOB35409 DFHPA1927 CICSIVP1 TRTABSZ=64, 40000200  
16.24.15 JOB35409 DFHPA1927 CICSIVP1 APPLID=CICSIVP1, 40000300  
16.24.15 JOB35409 DFHPA1927 CICSIVP1 CICSVC=233, 40000400  
16.24.15 JOB35409 DFHPA1927 CICSIVP1 .END 70000000  
16.24.15 JOB35409 DFHPA1103 CICSIVP1 END OF FILE ON SYSIN.  
16.24.16 JOB35409 +DFHTR0103 TRACE TABLE SIZE IS 64K  
16.24.16 JOB35409 +DFHSM0122I CICSIVP1 Limit of DSA storage below 16MB is 5,120K. 2  
16.24.16 JOB35409 +DFHSM0123I CICSIVP1 Limit of DSA storage above 16MB is 30M.  
16.24.16 JOB35409 +DFHSM0113I CICSIVP1 Storage protection is not active.  
16.24.16 JOB35409 +DFHSM0126I CICSIVP1 Transaction isolation is not active.  
16.24.16 JOB35409 +DFHDM0101I CICSIVP1 CICS is initializing.  
16.24.17 JOB35409 +DFHSI1500 CICSIVP1 CICS startup is in progress for CICS Transaction Server Version 3.2.0  
16.24.17 JOB35409 +DFHDH0100I CICSIVP1 Document domain initialization has started.  
16.24.17 JOB35409 +DFHEJ0101 CICSIVP1 747  
747 Enterprise Java domain initialization has started. Java is a  
747 trademark of Sun Microsystems, Inc.  
16.24.17 JOB35409 +DFHLG0101I CICSIVP1 Log manager domain initialization has started.  
16.24.17 JOB35409 +DFHRX0100I CICSIVP1 RX domain initialization has started.  
16.24.17 JOB35409 +DFHSO0100I CICSIVP1 Sockets domain initialization has started.  
16.24.17 JOB35409 +DFHWB0109I CICSIVP1 Web domain initialization has started.  
16.24.17 JOB35409 +DFHXS1100I CICSIVP1 Security initialization has started.  
16.24.17 JOB35409 +DFHRX0101I CICSIVP1 RX domain initialization has ended.  
16.24.17 JOB35409 +DFHSI1501I CICSIVP1 Loading CICS nucleus.  
16.24.17 JOB35409 +DFHDU0304I CICSIVP1 Transaction Dump Data set DFHDMPA opened.  
16.24.17 JOB35409 +DFHXS1102I CICSIVP1 Security is inactive. 3  
16.24.19 JOB35409 +DFHTR0113 CICSIVP1 Auxiliary trace is being started on data set DFHAUXT.  
16.24.19 JOB35409 +DFHCQ0100I CICSIVP1 Console queue initialization has started.  
16.24.19 JOB35409 +DFHCQ0101I CICSIVP1 Console queue initialization has ended.  
16.24.19 JOB35409 +DFHCQ0103I CICSIVP1 MVS console queue is open.  
16.24.19 JOB35409 +DFHCQ0200I CICSIVP1 CEKL transaction enabled.  
16.24.19 JOB35409 +DFHXS1101I CICSIVP1 Security initialization has ended.  
16.24.19 JOB35409 +DFHRM0140 CICSIVP1 Recovery manager autostart override found with value: 'AUTOINIT'.  
16.24.19 JOB35409 +DFHRM0149I CICSIVP1 Recovery manager autostart override record will be deleted.  
16.24.19 JOB35409 +DFHDH0101I CICSIVP1 Document domain initialization has ended.  
16.24.19 JOB35409 +DFHSO0101I CICSIVP1 Sockets domain initialization has ended.  
16.24.19 JOB35409 +DFHMN0105I CICSIVP1 Using default Monitoring Control Table.  
16.24.19 JOB35409 +DFHWB0110I CICSIVP1 Web domain initialization has ended.  
16.24.19 JOB35409 IEC031I D37-04,IFG0554P,DFHIVPBT,CICS,DFHAUXT,D306,P2P0C6,INST.CICSTS32.CICS.DFHAUXT  
16.24.19 JOB35409 +DFHTR0110 - AUXILIARY TRACE DATA SET DFHAUXT FULL - SWITCHING TO DFHBUXT  
16.24.19 JOB35409 +DFHMN0110I CICSIVP1 CICS Monitoring is inactive.  
16.24.19 JOB35409 IEC031I D37-04,IFG0554P,DFHIVPBT,CICS,DFHBUXT,D50B,P2P14B,INST.CICSTS32.CICS.DFHBUXT  
16.24.19 JOB35409 +DFHTR0109 - AUXILIARY TRACE DATA SET DFHBUXT FULL - AUXILIARY TRACE HAS BEEN STOPPED  
16.24.19 JOB35409 +DFHSI1502I CICSIVP1 CICS startup is Initial.  
16.24.19 JOB35409 +DFHTS0100I CICSIVP1 Temporary Storage initialization has started.  
16.24.19 JOB35409 +DFHLG0102I CICSIVP1 Log manager domain initialization has ended.  
16.24.19 JOB35409 +DFHSI1503I CICSIVP1 Terminal data sets are being opened.  
16.24.19 JOB35409 +DFHSI1592 CICSIVP1 CICS applid not (yet) active to VTAM.  
16.24.19 JOB35409 +DFHSI1572 CICSIVP1 Unable to OPEN VTAM ACB - RC=00000008, ACB Code=5A.  
16.24.19 JOB35409 +DFHKE0406I CICSIVP1 786  
786 CICS is about to wait for predecessors defined in the MVS automatic

**Notes:**

1 For information about the system initialization parameters used by the IVP jobs, see page “Specify system initialization parameters for the IVP jobs” on page 327. (See also 2 below.)

2 The DFHSM0122 and DFHSM0123 messages inform you of the limits available for the dynamic storage areas below and above 16MB. For information about these storage areas, see the *CICS System Definition Guide*.

**Note:** Storage for the extended read-only DSA, ERDSA, is obtained from read-only key 0 protected storage, because the default SIT specifies RENTPGM=PROTECT.

3 The DFHIVPBT job was run without external security active, because SEC=NO is specified as a SIT override parameter.

4 The default group list, DFHLIST, is used for this run of the DFHIVPBT job. Non-default functions (for example, the CICS online messages facility) are not available, because their CICS resources are not defined in this group list.

5 These messages are issued when CICS is initialized and the log streams do not exist. CICS issues a request to create the log stream dynamically using MVS define log stream services.

6 If system log initialization fails, CICS abends. (See also 1 in Figure 30 on page 334.)

7 If you want COBOL, C, C++ and PL/I languages, remove the comment marks from the SCEERUN and SCEERUN2 libraries, and increase the memory size for the job.

**Output from the DFHIVPBT job**

Output from the DFHIVPBT job (see Figure 30 on page 334) includes CICS messages written to one of the extrapartition destinations, responses to the transactions in the DFH\$BTCH data set, and an auxiliary trace.



**Notes:**

- 1 CICS messages issued when the log stream is created.
- 2 This message is sent to the CRDI destination.

## Verify interactive job, DFHIVPOL

The verify interactive job, DFHIVPOL, is tailored to your CICS environment and stored in the *hlq.XDFHINST* library when you run the DFHISTAR job. You can use the DFHIVPOL job to start up a CICS region and try out the current facilities; for example you can use the master terminal transaction, CEMT, and the resource definition transaction, CEDA. You can also run some CICS sample application programs (for example, the FILEA applications).

You need either an IBM 3270 Information Display System terminal or a **console device**. You can use both if you wish.

If you use an IBM 3270 Information Display System terminal with this IVP, you can try CEDA, CEMT and the sample applications.

From a console device, the CEDA transaction can be used only to INSTALL resource definitions. The sample programs cannot be executed from a console device.

If you want to communicate with CICS from an MVS console, you must define a console in the CSD before starting the IVP. (You cannot define a console in the TCT.) For more information, see “Defining an MVS console” on page 338.

If you want to communicate with CICS from a TSO session, you must define the TSO user as a console device in the CSD before starting the IVP. For more information, see “Defining a TSO user as a console device” on page 338.

The DFHIVPOL job invokes the CICS initialization program DFHSIP to start up CICS. The DFHSIP program reads system initialization parameters from the DFH\$SIP1 member of the SYSIN data set.

TCT=NO is specified as a SIT override which causes CICS to use the dummy terminal control table, DFHTCTDY. This dummy TCT contains only the CICS and VTAM control blocks that you need for use with VTAM terminals: there are no terminal entries.

For information about system initialization parameters specified as overrides for the run of the DFHIVPOL job, see “Specify system initialization parameters for the IVP jobs” on page 327.

### Defining a terminal for the online IVP

You can define a VTAM terminal by either of the following two methods:

1. Use the autoinstall facility, which is the recommended method, avoiding the need to define terminals to CICS explicitly before they can be used.
2. Define a terminal explicitly in the CSD, using the DEFINE command of DFHCSDUP, the batch utility for updating the CSD.

**Using autoinstall for a VTAM terminal:** If you use the autoinstall function of CICS, you avoid the need for each VTAM terminal that requires access to CICS being explicitly defined in the CSD. With autoinstall, the resource definitions you

create using RDO can act as models or templates for many resources of the same type. You then leave CICS to match real resources with one of the models. CICS installs table entries for these real resources dynamically, as and when they are needed.

When using autoinstall, you should be aware that when CICS processes an autoinstall request, it uses data from the VTAM logmode table. This is an important consideration. An autoinstall request will succeed only when the logmode data (which is passed to CICS in the BIND image) matches one of the model terminal definitions recorded in the autoinstall model table (AMT) from the CSD. For programming information about the LOGMODE definitions that match the CICS-supplied model definitions for autoinstall, see the *CICS Customization Guide*. Before attempting to start CICS and autoinstall a terminal for this IVP, check your VTAM definitions with those given in the *CICS Customization Guide*. If CICS fails to match model and logmode data, you receive message DFHZC6987I. For information about the suggested course of action if you receive message DFHZC6987I, see the *CICS Messages and Codes* manual.

*CSD resource definitions for autoinstall:* The CSD is defined and initialized for all the IVP jobs when you run the DFHCOMDS job (see “Creating the CICS data sets” on page 218), and includes some IBM-supplied definitions for use with autoinstall. These are defined in the following groups:

**Group Name**

**Description**

**DFHTERM**

Model terminal definitions for use with the autoinstall facility. For example, two of the TERMINAL definitions are 3270 and LU2.

**DFHTYPE**

Partial terminal definitions (TYPETERMs) defining common terminal properties, or attributes. For example, two of the TYPETERM definitions are DFH3270 (to define a non-SNA 3270 terminal) and DFHLU2E2 (to define a SNA 3270 model 2 terminal). The DFHLU2E2 resource definition matches the VTAM-supplied logmode SNX32702.

The DFHTERM and DFHTYPE groups are included in the CICS-defined group list called DFHLIST, which is defined in the GRPLIST operand in the sample SIT. If the CICS-supplied definitions are not suitable for your installation, you can create additional TYPETERM and model TERMINAL definitions in the CSD, but without a terminal you will have to do this offline, using the DFHCSDUP utility program. For information about autoinstall definitions, see the *CICS Resource Definition Guide*

Autoinstall also requires a user program to assign terminal identifiers, and, if necessary, to control access to the system. When you run the online IVP, you are unlikely to have any special requirements for terminal identifiers, or to control access, in which case you can use the IBM-supplied autoinstall user program, DFHZATDX. (If you are using autoinstall for APPC connections and terminals, the sample autoinstall user program is called DFHZATDY.)

**Defining a VTAM terminal in the CSD:** If you want to use an explicitly defined terminal, rather than let CICS autoinstall a terminal, you will need to define it offline using the DFHCSDUP utility program. The normal way to create resource definitions in the CSD is to use the CEDA DEFINE command from a CICS master terminal, but without a terminal you can only do this using the DFHCSDUP utility program. For an example of a DFHCSDUP job to define a VTAM terminal in the CSD, see Figure 31 on page 337. For information about the keywords and operands of the

DFHCSDUP DEFINE commands, see the *CICS Resource Definition Guide*.

```
//DEFTERM JOB (accounting information),MSGCLASS=A,
//      MSGLEVEL=(1,1),CLASS=A,NOTIFY=userid
//VTAMDEF EXEC PGM=DFHCSDUP
//STEPLIB DD DSN=CICSTS32.CICS.SDFHLOAD,DISP=SHR
//DFHCSD DD DSN=CICSTS32.CICS.DFHCSD,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
*
DEFINE TERMINAL(trmidnt) NETNAME(vtamname) GROUP(grpname)
      TYPETERM(name) INSERVICE(NO) AUTINSTMODEL(NO)
*
APPEND LIST(DFHLIST) TO(yourlist)
*
ADD GROUP(grpname) LIST(yourlist)
*
LIST LIST(yourlist) OBJECTS
/*
//
```

Figure 31. Defining a terminal by using the DFHCSDUP utility program

You must substitute your own values for the operands that are coded in lowercase in the DEFTERM job shown in Figure 31:

#### **TYPETERM**

Specify a unique name to identify the resource definition that matches the properties of the type of terminal you are using. For example, to define a SNA 3270 model 2 terminal, specify the CICS-supplied TYPETERM definition DFHLU2E2. For a list of the CICS-supplied TYPETERM definitions, or for information about creating your own definitions, see the *CICS Resource Definition Guide*.

#### **GROUP**

Code a unique name for the group to which the terminal resource definition is to belong.

#### **TERMINAL**

Code a unique 4-character terminal identifier as the name by which CICS is to know the terminal.

#### **NETNAME**

Code the 8-character VTAM name that identifies this terminal to your VTAM system.

#### **TO(yourlist) and LIST(yourlist)**

Code a unique name for *yourlist*. If your new group list does not include all the CICS-supplied resources as well as your own, you must specify DFHLIST and *yourlist* on the GRPLIST system initialization parameter of your CICS startup job.

To include the CICS-supplied list of resources in a new group list, create a new list by copying the CICS-supplied list, DFHLIST, using the APPEND command. (The CICS-supplied group list, DFHLIST, is a protected group that you cannot modify.) You can then add your resource definition groups to the new list. Before you run the IVP, make sure you define your new group list to CICS, by adding a SIT override to the SYSIN data set in the DFHIVPOL job stream.

## Defining the CICS APPLID to VTAM

You must ensure that either:

- VTAM knows the CICS application identifier (APPLID)  
or
- You change the CICS APPLID to one that is already known to your VTAM system.

If you use the default APPLID (DBDCCICS), define this to VTAM as described in “VTAM APPL parameters for CICS regions” on page 117, before starting the DFHIVPOL job.

## Defining an MVS console

If you want to use an MVS console with the DFHIVPOL job, CICS requires an installed definition for the console. You can achieve this using one of the following:

- An autoinstall model definition, in conjunction with autoinstall support for consoles. The model definition can specify any CONSNAME value, and references a TYPETERM definition that specifies DEVICE(CONSOLE).
- A predefined TERMINAL resource definition for a console, which specifies the console name on the CONSNAME attribute, and references a TYPETERM definition that specifies DEVICE(CONSOLE).

You define these resources using the DFHCSDUP utility program. The CICS-supplied TYPETERM group, DFHTYPE, contains a typeterm definition called DFHCONS, which is predefined with the required console properties. The DFHTERM group, however, does not contain any corresponding terminal entries for MVS consoles. You identify the console by the CONSNAME(*name*) attribute, even if the TERMINAL definition is an autoinstall model (the console name on an autoinstall model is a dummy value, and replaced by the real console name at install-time).

For an example of the DEFINE command required to define a console, see Figure 32 on page 339.

For information about defining MVS consoles to CICS, see the *CICS System Definition Guide*.

## Defining a TSO user as a console device

A TSO user can enter MODIFY commands from terminals logged on to TSO, using either the TSO CONSOLE command or from SDSF. MVS activates a console using, by default, the user's TSO user ID as the console name. To CICS, the console name passed on the MODIFY command is treated like an MVS system console, and requires an entry in the CICS system definition (CSD) file.

As in the case of the MVS system console, you can achieve this using one of the following:

- An autoinstall model definition, in conjunction with autoinstall support for consoles. The model definition can specify any CONSNAME value, and references a TYPETERM definition that specifies DEVICE(CONSOLE)
- A predefined TERMINAL resource definition for a console, which specifies the console name on the CONSNAME attribute, and references a TYPETERM definition that specifies DEVICE(CONSOLE).

You are recommended to define consoles to CICS with preset terminal security, using the USERID attribute on the TERMINAL definition. This avoids the TSO user



having to sign on using the CESN transaction. Otherwise, the TSO user's CICS signon password is displayed when entered on the CESN transaction.

For an example of a DEFINE command to define a TSO user, see Figure 32.

```
//DEFTERM JOB (accounting information),MSGCLASS=A,
//          MSGLEVEL=(1,1),CLASS=A,NOTIFY=userid
//CONSDEF EXEC PGM=DFHCSDUP
//STEPLIB DD DSN=CICSTS32.CICS.SDFHLOAD,DISP=SHR
//DFHCSD  DD DSN=CICSTS32.CICS.DFHCSD,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN   DD *

* Define a console
DEFINE TERMINAL(trmidnt)  GROUP(grpname)      TYPETERM(DFHCONS)
                          CONSNAME(consname)  DESCRIPTION(MVS CONSOLE consname)
                          USERID(tsouser)

* Define a TSO user as a console device
DEFINE TERMINAL(trmidnt)  GROUP(grpname)      TYPETERM(DFHCONS)
                          CONSNAME(tsouser)  DESCRIPTION(TSO USER tsouser)
                          USERID(tsouser)

*

APPEND LIST(DFHLIST) TO(yourlist)
*
ADD GROUP(grpname) LIST(yourlist)
*
LIST LIST(yourlist) OBJECTS
/*
//
```

Figure 32. Defining a console and a TSO user by using the DFHCSDUP utility program

**Note:** Substitute your own values for the operands that are shown in italics in the DEFTERM job shown in Figure 32.

To include the CICS-supplied list of resources at startup, specify DFHLIST on the GRPLIST system initialization parameter, as well as your own group list name. For example, specify GRPLIST=(DFHLIST,*userlista*,*userlistb*) in the CICS SYSIN data set member.

### Running the DFHIVPOL job

The DFHIVPOL job includes a procedure, DFHSTART, to start up CICS. When you have successfully logged on to CICS, you can carry out any of the interactive operations described in section “Using CICS-supplied transactions through a terminal” on page 341.

While logged on to CICS, you should perform a CEMT SET DUMPDS SWITCH to ensure that both dump data sets are initialized, before the dump utility program is run when you shut down CICS.

Finally, you can shut down CICS.

**Sample job log for the DFHIVPOL job:** When you run the DFHIVPOL job, your job log should look similar to the sample log shown in Figure 33 on page 340.

0

```

17.17.29 JOB35727 ---- TUESDAY, 24 APRIL 2007 ----
17.17.29 JOB35727 ICH70001I CICINST LAST ACCESS AT 16:24:15 ON TUESDAY, APRIL 24, 2007
17.17.29 JOB35727 $HASP373 DFHIVPOL STARTED - INIT 69 - CLASS A - SYS MV26
17.17.29 JOB35727 IEF403I DFHIVPOL - STARTED - TIME=17.17.29
17.17.29 JOB35727 - --TIMINGS (MINS.)-- ----PAGING COUNTS---
17.17.29 JOB35727 -JOBNAME STEPNAME PROCSTEP RC EXCP CPU SRB CLOCK SERV PG PAGE SWAP VIO SWAPS STE
17.17.29 JOB35727 -DFHIVPOL CICS CICSCNTL 01 23 .00 .00 .00 167 0 0 0 0 0
17.17.29 JOB35727 -DFHIVPOL CICS DTCNTL 01 19 .00 .00 .00 147 0 0 0 0 0
17.17.30 JOB35727 DFHPA1101 CICSIVP1 DFHSIT IS BEING LOADED.
17.17.30 JOB35727 DFHPA1108 CICSIVP1 DFHSIT HAS BEEN LOADED. (GENERATED AT: MM/DD= 01/12 HH:MM= 13:57).
17.17.30 JOB35727 DFHPA1100 CICSIVP1 OVERRIDE PARAMETERS FROM JCL EXEC STATEMENT: START=AUTO,SYSLIN
17.17.30 JOB35727 DFHPA1102 CICSIVP1 OVERRIDE PARAMETERS FROM SYSIN: 1
17.17.30 JOB35727 DFHPA1927 CICSIVP1 XRF=NO,
17.17.30 JOB35727 DFHPA1927 CICSIVP1 AUXTR=ON,
17.17.30 JOB35727 DFHPA1927 CICSIVP1 AUXTRSW=NEXT,
17.17.30 JOB35727 DFHPA1927 CICSIVP1 APPLID=CICSIVP1, 2
17.17.30 JOB35727 DFHPA1927 CICSIVP1 FCT=NO,
17.17.30 JOB35727 DFHPA1927 CICSIVP1 TCT=NO,
17.17.30 JOB35727 DFHPA1927 CICSIVP1 SRT=NO,
17.17.30 JOB35727 DFHPA1927 CICSIVP1 SEC=NO,
17.17.30 JOB35727 DFHPA1927 CICSIVP1 TRTABSZ=64,
17.17.30 JOB35727 DFHPA1927 CICSIVP1 PGRET=P/,
17.17.30 JOB35727 DFHPA1927 CICSIVP1 PGPURGE=T/,
17.17.30 JOB35727 DFHPA1927 CICSIVP1 PGCOPY=C/,
17.17.30 JOB35727 DFHPA1927 CICSIVP1 PGCHAIN=X/,
17.17.30 JOB35727 DFHPA1927 CICSIVP1 CICSSVC=233,
17.17.30 JOB35727 DFHPA1927 CICSIVP1 .END
17.17.30 JOB35727 DFHPA1103 CICSIVP1 END OF FILE ON SYSIN.
17.17.31 JOB35727 +DFHTR0103 TRACE TABLE SIZE IS 64K
17.17.31 JOB35727 +DFHSM0122I CICSIVP1 Limit of DSA storage below 16MB is 5,120K. 3
17.17.31 JOB35727 +DFHSM0123I CICSIVP1 Limit of DSA storage above 16MB is 30M.
17.17.31 JOB35727 +DFHSM0113I CICSIVP1 Storage protection is not active.
17.17.31 JOB35727 +DFHSM0126I CICSIVP1 Transaction isolation is not active.
17.17.32 JOB35727 +DFHDM0101I CICSIVP1 CICS is initializing.
17.17.32 JOB35727 +DFHWP0109I CICSIVP1 Web domain initialization has started.
17.17.32 JOB35727 +DFHSO0100I CICSIVP1 Sockets domain initialization has started.
17.17.32 JOB35727 +DFHRX0100I CICSIVP1 RX domain initialization has started.
17.17.32 JOB35727 +DFHRX0101I CICSIVP1 RX domain initialization has ended.
17.17.33 JOB35727 +DFHLG0101I CICSIVP1 Log manager domain initialization has started.
17.17.33 JOB35727 +DFHEJ0101 CICSIVP1 296
296 Enterprise Java domain initialization has started. Java is a
296 trademark of Sun Microsystems, Inc.
17.17.33 JOB35727 +DFHDH0100I CICSIVP1 Document domain initialization has started.
17.17.33 JOB35727 +DFHXS1100I CICSIVP1 Security initialization has started.
17.17.33 JOB35727 +DFHSI1500 CICSIVP1 CICS startup is in progress for CICS Transaction Server Version 3.2.0
17.17.33 JOB35727 +DFHDM0304I CICSIVP1 Transaction Dump Data set DFHMPA opened.
17.17.33 JOB35727 +DFHXS1102I CICSIVP1 Security is inactive.
17.17.33 JOB35727 +DFHSI1501I CICSIVP1 Loading CICS nucleus.
17.17.34 JOB35727 +DFHTR0113 CICSIVP1 Auxiliary trace is being started on data set DFHAUXT.
17.17.34 JOB35727 +DFHCQ0100I CICSIVP1 Console queue initialization has started.
17.17.34 JOB35727 +DFHCQ0101I CICSIVP1 Console queue initialization has ended.
17.17.34 JOB35727 +DFHCQ0103I CICSIVP1 MVS console queue is open.
17.17.34 JOB35727 +DFHCQ0200I CICSIVP1 CEKL transaction enabled.
17.17.34 JOB35727 +DFHXS1101I CICSIVP1 Security initialization has ended.
17.17.34 JOB35727 +DFHRM0141 CICSIVP1 Recovery manager autostart override record is not present.
Normal processing continues.
17.17.34 JOB35727 +DFHDH0101I CICSIVP1 Document domain initialization has ended.
17.17.34 JOB35727 +DFHMN0105I CICSIVP1 Using default Monitoring Control Table.
17.17.34 JOB35727 +DFHMN0110I CICSIVP1 CICS Monitoring is inactive.
17.17.34 JOB35727 +DFHSO0101I CICSIVP1 Sockets domain initialization has ended.
17.17.35 JOB35727 IEC031I D37-04,IFG0554P,DFHIVPOL,CICS,DFHAUXT,D306,P2P0C6,INST.CICSTS32.CICS.DFHAUXT
17.17.35 JOB35727 +DFHTR0110 - AUXILIARY TRACE DATA SET DFHAUXT FULL - SWITCHING TO DFHBUXT
17.17.35 JOB35727 +DFHWP0110I CICSIVP1 Web domain initialization has ended.
17.17.35 JOB35727 IEC031I D37-04,IFG0554P,DFHIVPOL,CICS,DFHBUXT,D50B,P2P14B,INST.CICSTS32.CICS.DFHBUXT
17.17.35 JOB35727 +DFHTR0109 - AUXILIARY TRACE DATA SET DFHBUXT FULL - AUXILIARY TRACE HAS BEEN STOPPED
17.17.35 JOB35727 +DFHSI1502I CICSIVP1 CICS startup is Warm.
17.17.35 JOB35727 +DFHTS0100I CICSIVP1 Temporary Storage initialization has started.
17.17.35 JOB35727 +DFHLG0103I CICSIVP1 System log (DFHLOG) initialization has started.
17.17.35 JOB35727 +DFHLG0104I CICSIVP1 322
322 System log (DFHLOG) initialization has ended. Log stream
322 CICINST.CICSIVP1.DFHLOG is connected to structure LOG GENERAL 008.

```

**Notes:**

1 For information about the system initialization parameters used by the IVP jobs, see page “Specify system initialization parameters for the IVP jobs” on page 327. (See also 2 and 3 below.)

2 For more information about defining an applid for the CICS IVP jobs, see Chapter 14, “Defining CICS regions as applications to VTAM,” on page 117. An applid of CICSIVP1 has been used in Figure 33 on page 340 page=no.

3 The DFHSM0122 and DFHM0123 messages inform you of the limits available for the dynamic storage areas below and above 16MB. For information about these storage areas, see the *CICS Performance Guide*.

**Note:** Storage for the extended read-only DSA, ERDSA, is obtained from read-only key 0 protected storage, because the sample SIT specifies RENTPGM=PROTECT (the default).

4 The DFHTM1715 message is issued because the CICS region was shut down by the terminal user (with netname IYCWTC30) issuing a CEMT PERFORM SHUTDOWN command.

7 If you want COBOL, C, C++ and PL/I languages, remove the comment marks from the SCEERUN and SCEERUN2 libraries, and increase the memory size for the job.

**Logging on at a VTAM terminal**

When the DFHIVPOL job displays the console message CONTROL IS BEING GIVEN TO CICS, you can log on to CICS using an IBM 3270 Information Display system terminal. Use the CICS application identifier that you specified when you brought up CICS to log on through your VTAM terminal. For example, unless you changed the APPLID specified as a SIT override parameter, (it is CICSIVP1), enter LOGON APPLID(CICSIVP1).

If you are using autoinstall, your logon request is passed to CICS and, provided all the autoinstall requirements described in “Using autoinstall for a VTAM terminal” on page 335 have been met, CICS installs your terminal. It does this by creating a TCT terminal entry (TCTTE) using the model definitions defined in the group list, DFHLIST, and the terminal identifier returned by the autoinstall user program (DFHZATDX in this case).

If you are using a terminal defined in the CSD explicitly, and included in the group list specified in the startup job stream, CICS identifies the installed resource definitions by the VTAM net name, and creates the required TCTTE.

When you log onto CICS, your terminal can display a “good morning” message, by the transaction specified on the GMTRAN system initialization parameter. The default transaction, CSGM, displays a welcome message as defined by the GMTEXT system initialization parameter.

**Using CICS-supplied transactions through a terminal**

After you have started CICS with the DFHIVPOL job, you can use the CICS-supplied transactions to try out various functions of CICS, to help you verify that CICS is working properly. You can use the transactions at a CICS terminal and, if you defined one, the system console.

Table 26 shows some typical terminal interactions, including use of the CEMT transaction. For information about the CICS transactions that you can try with the DFHIVPOL job, and about the message-switching responses to those transactions, see the *CICS Supplied Transactions* manual.

Table 26. Typical terminal interactions

Operator Input	System Response
CEMT	Status: ENTER ONE OF THE FOLLOWING  Discard Inquire Perform Set
I	Status: ENTER ONE OF THE FOLLOWING OR HIT ENTER FOR DEFAULT (Followed by a list of options)
PROG Press ENTER key	STATUS: RESULTS - OVERTYPE TO MODIFY Prog(CEECBLDY) Len(0000000) Ass Pro Ena Pri Res(000) Use(0000000000) Any Cex Ful
Press PF3 key Press CLEAR key CEMT PERFORM STATISTICS	SESSION ENDED
Press PF3 key Press CLEAR key CETR	Clear or PF3 pressed Normal termination of CETR
CEMT I TA	Displays list of tasks in the system SESSION ENDED
Press PF3 key Press CLEAR key	
CEMT I PROG(DFHFEP)	Prog(DFHFEP )Len(005848) Ass Pro Ena Pri Res(000) Use(0000000) Any Cex Ful Qua
Press PF3 key Press CLEAR key	SESSION ENDED
CEOT (Inquire about this terminal)	Ter (tmid) Tra (CEOT) Pri (nnn) Pag Ins Ati Tti (Remember 'tmid' for use in next transaction, CMSG)
Press PF3 key Press CLEAR key	SESSION ENDED

Table 26. Typical terminal interactions (continued)

Operator Input	System Response
CMSG 'HELLO',R=tmid,S	(Send the message 'HELLO' to your terminal) MRS OK MESSAGE HAS BEEN ROUTED (briefly at bottom right of screen) HELLO (at top left of screen)

You may enter your CEMT input in either uppercase or lowercase, because the master terminal transaction translates all input to uppercase. Use the CLEAR key and the PF3 key as indicated.

If you enter the CETR transaction, CICS displays the status of the various trace options. For information about the CETR transaction, and the other information panels available by using specified PF keys, see the *CICS Supplied Transactions* manual.

You can alter the status of any of the trace options by overtyping the current value, indicated by ===> on the CETR display.

### Using the CEDA transaction

When DFHVPOL starts up CICS, it uses the unsuffixed SIT, DFHSIT. This system initialization table specifies GRPLIST=DFHLIST, causing all the CICS resource definitions that are needed for normal running to be installed.

You can see which resources are included in DFHLIST by using the CEDA transaction, for example, by using the **CEDA EXPAND LIST(DFHLIST)** command.

Press PF8 to see the continuation of the list. If you started the DFHVPOL job with your own group list specified instead of the DFHLIST group list, specify the name of your list in the CEDA EXPAND command. The CICS-defined groups all begin with DFH. For information about CEDA and the interactions for a typical sequence of CEDA commands, see the *CICS Resource Definition Guide*.

The DFHLIST group list does not include any of the sample applications groups, the group names of which all begin with DFH\$. To use the sample programs, therefore, you must first install the resource definitions for the required samples. For example, to use the FILEA sample application:

1. Install the sample programs that are needed for the FILEA applications. You can do this by the command:  
CEDA INSTALL GROUP(DFH\$AFLA)
2. Make the FILEA data set available to CICS. You can do this by one of the following:
  - Install a FILE resource definition for the FILEA data set. You can do this by the command:  
CEDA INSTALL GROUP(DFH\$FILA)
  - Provide a DD statement for the FILEA data set in your CICS startup JCL. For example,  
//FILEA DD DISP=SHR,DSN=CICSTS32.CICS.CICSHTH1.FILEA

To end the CEDA session, press PF3.

## Invoking and executing sample programs

To try the assembler-language version of the FILEA sample application, install group DFH\$AFLA then enter the AMNU transaction.

## Using transactions from a console device

CICS transactions (other than CECI) can be invoked from a console device, and other CICS operators can communicate with the console operator. In particular, you can use the console device for CICS master terminal functions, to control CICS terminals or to control several CICS regions in conjunction with multiregion operation. Normal operating-system use of the console device is not inhibited, and CICS supports multiple console devices where present.

### Note:

1. The CEDA transaction can be used from a console device only to INSTALL resource definitions.
2. The CECI transaction and the sample programs cannot be used from a console device.

If you issue the MVS command `d consoles`, this displays a list of console devices. This list identifies the console devices by name.

You can use a console device to submit MODIFY commands from your job stream if you define a console device in your CSD as `CONSNAME(INTERNAL)`.

For further information about defining consoles, see “Defining an MVS console” on page 338. For further information about defining TSO users as consoles, see “Defining a TSO user as a console device” on page 338.

To enter a command, use:

```
{MODIFY|F} jobname, ['command']
```

where:

### jobname

is the region identifier for the CICS region. This is either the name of the job being used to execute CICS (for example, DFHIVPOL) or the name of a procedure if CICS was initiated as a started task.

### command

is a string of data, starting with a CICS transaction identifier. If the transaction requires further input, the operator is prompted in the same way as any normal terminal operator. The message from CICS contains a reply number that must be quoted in the reply.

You can use the commands shown in Figure 34 on page 345 to verify the CEMT and CEOT transactions from the MVS console. (For information about these transactions, see the *CICS Supplied Transactions* manual.)

**Entering commands from TSO:** A TSO user can enter CICS commands as above after invoking the TSO command `CONSOLE`, in either of the following formats:

```
CONSOLE {MODIFY|F} cicsid, ['command']
```

```
CONSOLE  
{MODIFY|F} cicsid, ['command']  
END
```

When the TSO command `CONSOLE` is used, TSO checks the user for authority to issue console commands. Further, if console operator command security is active, the TSO user must be specifically authorized to issue `MODIFY cicsid`.

The TSO user can interact with an alternate CICS by using the command `CONSOLE MODIFY altcics,CEBT`.

You can also use TSO CLIST processing to issue sequences of CICS commands.

Operator Input	System Response
f dfhivpol,'cent i terminal'	Displays a list of terminals attached to CICS
f dfhivpol,'cent i dump'	Displays status of transaction dump data sets
f dfhivpol,'cent p statistics'	CICS writes statistics to SMF data sets
f dfhivpol,'cent i ta'	Displays number and types of tasks currently running
f dfhivpol,'cent p dump'	CICS invokes SDUMP macro for system dump to be taken
f dfhivpol,'cent i prog(dfhpep)'	Displays details of DFHPEP module
f dfhivpol,'ceot'	Displays details of operator console
f dfhivpol,'cent i journalname'	Displays status of CICS logs

Figure 34. Using an MVS console for master terminal operations

### Terminating CICS

To terminate CICS, enter: `CEMT P SHUT` from the VTAM terminal or MVS console. (This is a short form of `CEMT PERFORM SHUTDOWN`.) The system responds with message `DFH1713`, and those that follow, as shown in the sample job log shown on page Figure 33 on page 340.

### Verifying shared data tables support

To verify that the shared data tables function can be used, you can:

1. Start up a CICS region on which you have installed support for shared data tables.

**Note:** To use shared data tables, you must install the following modules: `DFHDTSVC`, `DFHDTCV`, and `DFHMVRMS` in either an authorized system library in the MVS linklist (LNKLIST concatenation of the MVS system) or in the LPA. When you install CICS, these modules are installed into the `hlq.SDFHLINK` library (which you should normally include in the MVS linklist).

2. Define and install a user-maintained data table.
3. Try a generic read command on your data table, using the `CECI` transaction. (Generic reads of user-maintained data tables are allowed only with shared data tables.) If shared data tables is operational, you should see a normal response. If shared data tables is not operational, you would see an `INVREQ` response.

**Note:** This verification process uses user-maintained data tables throughout, because the behavior of CICS-maintained data tables is transparent to

their users. For example, a normal response is returned for a generic read of a CICS-maintained data table, regardless of whether or not shared data tables is operational.

To verify that the cross-memory services of shared data tables are working:

4. Start up a second CICS region (the **requester**) that has an interregion communication (IRC) connection to the first CICS region (the **server**, which contains the user-maintained data table and source data set).
5. On the requester CICS region, do the following:
  - a. Define and install a remote file referring to (associated with) the user-maintained data table on the server CICS region.
  - b. Close the interregion communication connection between the two CICS regions so that function shipping is impossible; that is, only the cross-memory services of shared data tables can be used to access the shared data table from the requester CICS region. To close the connection, you can enter the command:

```
CEMT SET IRC CLOSED
```

To verify that function shipping cannot work, try a remote READ of a file (not a data table) on the server CICS region; you will get a SYSIDERR response.

- c. Try a generic read command on your data table, using the CECI transaction. If the cross-memory services of shared data tables can be used, you should see a normal response.
6. To restore interregion communication between the two CICS regions, open the connection again. To do this, you can enter the command:

```
CEMT SET IRC OPEN
```

### Example verification of shared data tables

As an example verification test of shared data tables, the following steps were completed for the CICS shared data tables environment shown in Figure 35 on page 347:

1. A CICS region, CICSIDC, was started. (CICSIDC is the server CICS region in this example.)
2. On CICSIDC, the following steps were completed:
  - a. The user-maintained data table, MYSDT, was defined and installed. The MYSDT data table was based on the sample data set, *hlq.CICSIDC.FILEA*, installed on that region.
  - b. The following generic READ command was entered at a terminal:  

```
CECI READ FILE(MYSDT) RIDFLD(00092) KEYLENGTH(5) GE GTEQ
```

 Figure 36 on page 348 shows the initial response (LOADING), and Figure 37 on page 348 shows the subsequent response when the command was repeated after the data table had completed loading.

The following steps were completed to verify the cross-memory services of shared data tables:

3. A second CICS region, CICSIDA, was started with support for shared data tables. (CICSIDA is the requester CICS region in this example.)
4. The following IRC connections and sessions were defined and installed on the associated CICS regions:

Region	CONNECTION	SESSION
CICSIDA	CICA	ATOC
CICSIDC	CICC	CTOA



See Figure 40 on page 350 and Figure 41 on page 350 for the parameters used for the CICA and ATOC resource definitions. The parameters for the CICC and CTOA resource definitions were similar.

5. On CICSIDA, the following steps were completed:
  - a. The file, REMSDT, was defined and installed as remote, referring to the MYSDT data table on CICSIDC. See Figure 42 on page 351 for the parameters used for the REMSDT resource definition.
  - b. The file, REMFIL, was defined and installed as remote, referring to the FILEA sample file on CICSIDC.
  - c. The CEMT SET IRC CLOSED command was used to close the IRC connection to CICSIDC.
  - d. The following generic READ command was entered at a terminal:

```
CECI READ FILE(REMFIL) RIDFLD(00092) KEYLENGTH(5)
LENGTH(80) GE GTEQ
```

Figure 38 on page 349 shows the response (SYSIDERR), because the remote file cannot be accessed by function-shipping. (This response would also be observed for the remote data table, REMSDT, if the IRC connection was closed.)

- e. The following generic READ command was entered at a terminal:

```
CECI READ FILE(REMSDT) RIDFLD(00092) KEYLENGTH(5)
LENGTH(80) GE GTEQ
```

Figure 39 on page 349 shows the response (NORMAL). This only works if MYSDT is already open on CICSIDC, as achieved by step 2b on page 346.

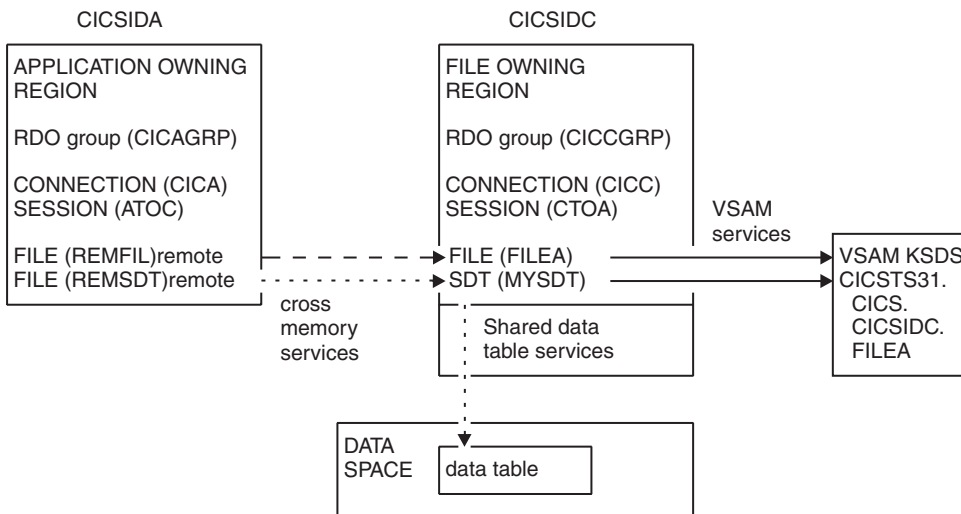


Figure 35. CICS environment for example verification of shared data tables

```

read file(MYSDT) ridfld(00092) keylength(5) ge gteq
STATUS:  COMMAND EXECUTION COMPLETE          NAME=
EXEC CICS READ
  File( 'MYSDT ' )
  < SYsid() >
  ( SEt() | Into( ' ' ) )
  < Length( +00000 ) >
  RIdfld( '00092' )
  < Keylength( +00005 ) < GGeneric > >
  < RBa | RRn | DEBRec | DEBKey >
  < GTeq | Equal >
  < UNcommitted | Consistent | REpeatable | UPdate <token()> >
  < Nosuspend >

```

```

RESPONSE: LOADING          EIBRESP=+0000000094 EIBRESP2=+0000000104
PF 1 HELP 2 HEX 3 END 4 EIB 5 VAR 6 USER 7 SBH 8 SFH 9 MSG 10 SB 11 SF

```

Figure 36. On CICSIDC, response to initial CECI generic READ FILE command with SDT support. (The data table is loaded on first reference, and generic READ commands are not allowed for a user-maintained data table while it is loading.)

```

read file(MYSDT) ridfld(00092) keylength(5) ge gteq
STATUS:  COMMAND EXECUTION COMPLETE          NAME=
EXEC CICS READ
  File( 'MYSDT ' )
  < SYsid() >
  ( SEt()
    | Into( ' 000983J. S. TILLING          WASHINGTON, DC          34512' ... ) )
  < Length( +00080 ) >
  RIdfld( '00092' )
  < Keylength( +00005 ) < GGeneric > >
  < RBa | RRn | DEBRec | DEBKey >
  < GTeq | Equal >
  < UNcommitted | Consistent | REpeatable | UPdate <token()> >
  < Nosuspend >

```

```

RESPONSE: NORMAL          EIBRESP=+0000000000 EIBRESP2=+0000000000
PF 1 HELP 2 HEX 3 END 4 EIB 5 VAR 6 USER 7 SBH 8 SFH 9 MSG 10 SB 11 SF

```

Figure 37. On CICSIDC, response to CECI generic READ FILE command with SDT support. Normal response

```

read file(FILEA) ridfld(00092) keylength(5) length(80) ge gteq
STATUS:  COMMAND EXECUTION COMPLETE          NAME=
EXEC CICS  READ
  File( 'FILEA  ' )
  < SYsid() >
  ( SEt()
    | Into( '                               ' ... ) )
  < Length( +00080 ) >
  RIdfld( '00092' )
  < Keylength( +00005 ) < GGeneric > >
  < RBa | RRn | DEBRec | DEBKey >
  < GTeq | Equal >
  < UNcommitted | Consistent | REpeatable | UDate <token()> >

  < Nosuspend >

```

```

RESPONSE: SYSIDERR          EIBRESP=+0000000053 EIBRESP2=+0000000130
PF 1 HELP 2 HEX 3 END 4 EIB 5 VAR 6 USER 7 SBH 8 SFH 9 MSG 10 SB 11

```

Figure 38. On CICSIDA, response to remote CECI generic READ FILE command, with IRC closed. SYSIDERR response for file, REMFIL, attempting to use function shipping for associated file, FILEA, on CICSIDC

```

read file(MYSDT) ridfld(00092) keylength(5) length(80) ge gteq
STATUS:  COMMAND EXECUTION COMPLETE          NAME=
EXEC CICS  READ
  File( 'MYSDT  ' )
  < SYsid() >
  ( SEt()
    | Into( ' 000983J. S. TILLING      WASHINGTON, DC      34512' ... ) )
  < Length( +00080 ) >
  RIdfld( '00092' )
  < Keylength( +00005 ) < GGeneric > >
  < RBa | RRn | DEBRec | DEBKey >
  < GTeq | Equal >
  < UNcommitted | Consistent | REpeatable | UDate <token()> >

  < Nosuspend >

```

```

RESPONSE: NORMAL          EIBRESP=+0000000000 EIBRESP2=+0000000000
PF 1 HELP 2 HEX 3 END 4 EIB 5 VAR 6 USER 7 SBH 8 SFH 9 MSG 10 SB 11 SF

```

Figure 39. On CICSIDA, response to remote CECI generic READ FILE command, with IRC closed. Normal response for file, REMSDT, using cross-memory services for associated shared data table, MYSDT, on CICSIDC

OBJECT CHARACTERISTICS CICS RELEASE = 0650

```
Connection      : CICA
Group           : CICAGRP
Description     : MRO CONNECTION CICSIDA TO CICSIDC
CONNECTION IDENTIFIERS
Netname        : CICSIDC
INDsys         :
REMOTE ATTRIBUTES
REMOTESystem   :
REMOTENAME     :
CONNECTION PROPERTIES
Accessmethod   : IRC          Vtam | IRC | INdirect | Xm
Protocol       :              Appc | Lu61
Singlesess    : No           No | Yes
Datastream     : User        User | 3270 | SCs | STRfield | Lms
RECORDformat   : U           U | Vb
OPERATIONAL PROPERTIES
Autoconnect    : No          No | Yes | All
INService      : Yes         Yes | No
```

Figure 40. Example CONNECTION resource definition, CICA, installed on CICSIDA. Only relevant parameters are shown; other parameters were allowed to default

OBJECT CHARACTERISTICS CICS RELEASE = 0650

```
Sessions       : ATOC
Group          : CICAGRP
Description    : SESSION FOR MRO CICA TO CICC
SESSION IDENTIFIERS
Connection     : CICA
SESSName      :
NETnameq      :
MODename      :
SESSION PROPERTIES
Protocol       : Lu61          Appc | Lu61
MAXimum       : 000 , 000      0-999
RECEIVEPfx    : RB
RECEIVECount  : 005           1-999
SENDPfx       : SB
SENDCount     : 003           1-999
SENDSize      : 04096         1-30720
RECEIVESize   : 04096         1-30720
SESSPriority   : 100           0-255
```

Figure 41. Example SESSION resource definition, ATOC, associated with connection, CICA. Only relevant parameters are shown; other parameters were allowed to default

## OBJECT CHARACTERISTICS

CICS RELEASE = 0650

```

File          : REMSDT
Group         : CICCGRP
DEscription  :
VSAM PARAMETERS
DSName       :
Password     :                PASSWORD NOT SPECIFIED
RLsaccess    : No             No | Yes
Lsrpoolid    : 1             1-8 | None
READInteg    : Uncommitted   Uncommitted | Consistent | Repeat
DSNSharing   : Allreqs       Allreqs | Modifyreqs
STRings      : 001           1-255
Nsrgroup     :
REMOTE ATTRIBUTES
REMOTESystem : CICC
REMOTENAME   : MYSDT
RECORDSize   :                1-32767
Keylength    :                1-255
INITIAL STATUS
STATus       : Enabled       Enabled | Disabled | Unenabled

```

Figure 42. Example remote FILE resource definition, REMSDT, installed on CICSIDA. Only relevant parameters are shown; other parameters were allowed to default

## Verifying the CICS-DBCTL interface

This section describes how to use the installation verification procedure, DFHIVPDB, which you can use to verify that the CICS-DBCTL interface can be used successfully.

Before you can run the DFHIVPDB job successfully, you must:

1. Tailor the DFHIVPDB job to your CICS and IMS environment.

You can do this as part of the process of tailoring all CICS sample post-installation jobs, as described in “Defining DL/I support” on page 230. When you run the DFHISTAR job as part of the CICS installation process, the DFHIVPDB job is installed in the *hlq.XDFHINST* library.

**Note:** Change the prefix of the IMS.SDFSRESL (previously called IMS.RESLIB) library in the DFHIVPDB job to the prefix that you use for your IMS libraries.

2. Create the data sets needed by the CICS region used by the the DFHIVPDB job.

To do this, you can tailor and run copies of the following CICS sample jobs:

### DFHCOMDS

This job creates the CICS data sets common to all CICS regions.

### DFHDEFDS

This job creates the data sets needed for each CICS region.

When you run the DFHISTAR job as part of the CICS installation process, these jobs are installed in the *hlq.XDFHINST* library.

3. Run the IMS installation verification procedures, as outlined in “The IMS installation requirements for the DFHIVPDB job” on page 352.

## The IMS installation requirements for the DFHIVPDB job

The DFHIVPDB job depends on you running the IMS installation verification procedures, as part of the INSTALL/IVP process described in the *IMS Installation Guide*. The following assumptions about the IMS INSTALL/IVP process are made:

1. The IMS sample database, DI21PART, has been successfully defined. This comprises two data sets:
  - DI21PART
  - DI21PARO
2. The DI21PART database has been loaded with the IMS-supplied sample data.
3. The following IMS-supplied procedures have been installed in an executable procedure library:
  - ACBGEN
  - PSBGEN
4. The sample DRA startup table, DFSPZPIV, has been built and installed in the IMS.SDFSRESL (previously called IMS.RESLIB) library.
5. The sample DBCTL system, IVP3, is available.

For information about installing IMS, the INSTALL/IVP process, and running the IMS IVPs, see the *IMS Installation Guide*.

## The DFHIVPDB job steps

The DFHIVPDB job consists of the following job steps:

1. **GEN.** This step unloads the member DFH\$DBAN from the *hlq.SDFHSAMP* library into a temporary sequential data set called CARDIN. This member contains the transactions to invoke the assembler versions of the DL/I sample applications that CICS reads from CARDIN as soon as initialization is complete.

**Note:** The sequential data set CARDIN is defined in the sample terminal control table, DFHTCT5\$, as a simulated terminal.

The COBOL version, DFH\$DBCBC, and the PL/I version, DFH\$DBPL, of the sample DL/I transactions are also in the *hlq.SDFHSAMP* library. If you want to run the COBOL or PL/I versions, modify this job step to load CARDIN with the appropriate member.

Output generated by the transactions is sent to a similar device – a sequential data set defined as PRINTER.

2. **CICS.** This job step executes the DFHSTART procedure to start up CICS, with the CICS-supplied resource group list DFH\$IVPL. CICS attempts to connect to the DBCTL system IVP3, run the sample DLI transactions, and then shutdown the CICS region.

**Note:** If the DBCTL system, IVP3, is not running, the sample DLI transactions will abend.

If you want to examine the sample members used by this IVP, here is a list of them, and where you can find each one:

### DFHIVPDB

This IVP contains some explanatory comments, and was installed in the *hlq.XDFHINST* library when you ran the DFHISTAR job. For details of the DFHISTAR job, see “Tailoring the CICS-supplied skeleton jobs” on page 215.

## DFH\$SIP5

This is the member of the *hlq*.SYSIN data set that contains the system initialization parameter overrides specific to the DFHIVPDB job.

**Note:** You will probably want to specify other system initialization parameters (for example, APPLID, CICSSVC, and DFLTUSER) for the DFHIVPDB job; the DFH\$SIP5 member of the *hlq*.SYSIN data set is a convenient place to do so.

## DFHTCT5\$

This is the sample TCT that specifies the sequential devices that CICS uses in this IVP as a simulated terminal, with a terminal name of SAMA. The source statements are in the member, DFH\$TCTS, of the *hlq*.SDFHSAMP library.

**Running the DFHIVPDB job:** Before submitting the DFHIVPDB job, run the DFHRMUTL program, as shown below, to reset the global catalog control record to perform an INITIAL start on the next CICS startup.

```
//DFHRMUTI JOB 24116475,'DFHRMUTL',
//          CLASS=A,MSGCLASS=H,NOTIFY=userid
//*
//*-----*/
//* RESET GLOBAL CATALOG CONTROL RECORD TO INITIAL START */
//*-----*/
//DFHRMUTL EXEC PGM=DFHRMUTL,REGION=1M
//STEPLIB DD DSN=CICSTS32.CICS.SDFHLOAD,DISP=SHR
//SYSPRINT DD SYSOUT=*
//DFHGCD DD DSN=CICSTS32.CICS.DBCCICX.DFHGCD,DISP=OLD
//SYSIN DD *
SET_AUTO_START=AUTOINIT
/*
```

When you are satisfied that you have made all the necessary preparations, and that all the prerequisite jobs have been run, submit the DFHIVPDB job. The job loads the DL/I transactions into CARDIN. CICS reads the transactions, and sends the output to the PRINTER sequential data set.

### Note:

1. The first transaction copied from the DFH\$DBAN member of the *hlq*.SDFHSAMP library to CARDIN is CDBC CONNECT SUFFIX(IV). This connects CICS to DBCTL, using the sample DRA startup table, DFSPZPIV.
2. The final transaction copied from the DFH\$DBAN member of the *hlq*.SDFHSAMP library to CARDIN is CEMT PERFORM SHUT.

If you want to use some commands online before CICS shuts down, then delete the CEMT command before you run the job. You will then be able to issue CEMT, CEDA and other CICS-supplied transactions, and initiate a shutdown either from a CICS terminal or through an MVS console. If you want to communicate with CICS through an MVS console, you must define a console to CICS before you start DFHIVPDB, as described in “Defining an MVS console” on page 338. If you want to enter MODIFY commands from terminals connected to TSO, you must define the TSO users as console devices, as described in “Defining a TSO user as a console device” on page 338.

A sample job log from a run of the DFHIVPDB job is given in Figure 43 on page 355. The results you get from the transaction processing should be similar to those shown in Figure 43 on page 355, Figure 44 on page 356, and Figure 45 on page 356

356.



1  
0

J E S 2 J O B L O G -- S Y S T E M M V 2 6 -- N O D E W I N M V S 2 C

```
09.36.19 JOB36923 ---- TUESDAY, 24 APR 2007 ----
09.36.19 JOB36923 ICH70001I CICINST LAST ACCESS AT 08:27:32 ON TUESDAY, APRIL 24, 2007
09.36.19 JOB36923 $HASP373 DFHIVPDB STARTED - INIT 4 - CLASS A - SYS MV26
09.36.19 JOB36923 IEF403I DFHIVPDB - STARTED - TIME=09.36.19
09.36.19 JOB36923 -
                                --TIMINGS (MINS.)--
                                -----PAGING COUNTS-----
09.36.19 JOB36923 -JOBNAME  STEPNAME  PROCSTEP   RC   EXCP   CPU   SRB   CLOCK   SERV  PG   PAGE   SWAP   VIO  SWAPS
09.36.19 JOB36923 -DFHIVPDB          GEN          00    53    .00    .00    .00    184   0     0     0     0     0
09.36.19 JOB36923 -DFHIVPDB  CICS    CICSCNTL  01    16    .00    .00    .00    148   0     0     0     0     0
09.36.19 JOB36923 -DFHIVPDB  CICS    DTCNTL   01    15    .00    .00    .00    161   0     0     0     0     0
09.36.20 JOB36923 DFHPA1101 CICSIVP1 DFHSIT IS BEING LOADED.
09.36.20 JOB36923 DFHPA1108 CICSIVP1 DFHSIT HAS BEEN LOADED. (GENERATED AT: MM/DD= 01/12 HH:MM= 13:57).
09.36.20 JOB36923 DFHPA1100 CICSIVP1 OVERRIDE PARAMETERS FROM JCL EXEC STATEMENT: START=AUTO,SYSIN
09.36.20 JOB36923 DFHPA1102 CICSIVP1 OVERRIDE PARAMETERS FROM SYSIN: 1
09.36.20 JOB36923 DFHPA1927 CICSIVP1 GRPLIST=DFH$IVPL, INCLUDE DLI SAMPLE PROGRAMS & TRANSACTIONS 20000000
09.36.20 JOB36923 DFHPA1927 CICSIVP1 FCT=NO, 30000000
09.36.20 JOB36923 DFHPA1927 CICSIVP1 TCT=5$, TCT INCLUDES SEQ DEVICES 40000000
09.36.20 JOB36923 DFHPA1927 CICSIVP1 XRF=NO, 50000000
09.36.20 JOB36923 DFHPA1927 CICSIVP1 STNTR=OFF, 53000000
09.36.20 JOB36923 DFHPA1927 CICSIVP1 STNTRFC=1, TRACE FILE CONTROL AND DLI EVENTS 56000000
09.36.20 JOB36923 DFHPA1927 CICSIVP1 AUXTR=ON, 60000000
09.36.20 JOB36923 DFHPA1927 CICSIVP1 AUXTRSW=NEXT, 70000000
09.36.20 JOB36923 DFHPA1927 CICSIVP1 SRT=NO, 70000100
09.36.20 JOB36923 DFHPA1927 CICSIVP1 SEC=NO, 70000200
09.36.20 JOB36923 DFHPA1927 CICSIVP1 TRTABSZ=64, 70000300
09.36.20 JOB36923 DFHPA1927 CICSIVP1 APPLID=CICSIVP1, 70000400
09.36.20 JOB36923 DFHPA1927 CICSIVP1 CICSSVC=233, 70000500
09.36.20 JOB36923 DFHPA1927 CICSIVP1 .END 80000000
09.36.20 JOB36923 DFHPA1103 CICSIVP1 END OF FILE ON SYSIN. 09.36.21 JOB36923 +DFHTR0103 TRACE TABLE SIZE IS 64K
09.36.21 JOB36923 +DFHSM0122I CICSIVP1 Limit of DSA storage below 16MB is 5,120K.
09.36.21 JOB36923 +DFHSM0123I CICSIVP1 Limit of DSA storage above 16MB is 30M.
09.36.21 JOB36923 +DFHSM0113I CICSIVP1 Storage protection is not active.
09.36.21 JOB36923 +DFHSM0126I CICSIVP1 Transaction isolation is not active.
09.36.21 JOB36923 +DFHDM0101I CICSIVP1 CICS is initializing.
09.36.23 JOB36923 +DFHQB0109I CICSIVP1 Web domain initialization has started.
09.36.23 JOB36923 +DFHQB0100I CICSIVP1 Sockets domain initialization has started.
09.36.23 JOB36923 +DFHRX0100I CICSIVP1 RX domain initialization has started.
09.36.23 JOB36923 +DFHRX0101I CICSIVP1 RX domain initialization has ended.
09.36.23 JOB36923 +DFHLG0101I CICSIVP1 Log manager domain initialization has started.
09.36.23 JOB36923 +DFHEJ0101 CICSIVP1 790
790 Enterprise Java domain initialization has started. Java is a
790 trademark of Sun Microsystems, Inc.
09.36.23 JOB36923 +DFHQB0100I CICSIVP1 Document domain initialization has started.
09.36.23 JOB36923 +DFHXS1100I CICSIVP1 Security initialization has started.
09.36.23 JOB36923 +DFHSI1500 CICSIVP1 CICS startup is in progress for CICS Transaction Server Version 3.2.0
09.36.23 JOB36923 +DFHXS1102I CICSIVP1 Security is inactive.
09.36.23 JOB36923 +DFHQB0304I CICSIVP1 Transaction Dump Data set DFHQBMPB opened.
09.36.23 JOB36923 +DFHSI1501I CICSIVP1 Loading CICS nucleus.
09.36.26 JOB36923 +DFHQB0113 CICSIVP1 Auxiliary trace is being started on data set DFHAUXT.
09.36.26 JOB36923 +DFHQB0100I CICSIVP1 Console queue initialization has started.
09.36.26 JOB36923 +DFHQB0101I CICSIVP1 Console queue initialization has ended.
09.36.26 JOB36923 +DFHQB0103I CICSIVP1 MVS console queue is open.
09.36.26 JOB36923 +DFHQB0200I CICSIVP1 CEKL transaction enabled.
09.36.26 JOB36923 +DFHXS1101I CICSIVP1 Security initialization has ended.
09.36.26 JOB36923 +DFHRM0140 CICSIVP1 Recovery manager autostart override found with value: 'AUTOINIT'.
09.36.26 JOB36923 +DFHRM0149I CICSIVP1 Recovery manager autostart override record will be deleted.
09.36.26 JOB36923 +DFHQB0101I CICSIVP1 Document domain initialization has ended.
09.36.26 JOB36923 +DFHQB0105I CICSIVP1 Using default Monitoring Control Table.
09.36.26 JOB36923 +DFHQB0101I CICSIVP1 Sockets domain initialization has ended.
09.36.26 JOB36923 +DFHQB0110I CICSIVP1 Web domain initialization has ended.
09.36.26 JOB36923 +DFHQB0110I CICSIVP1 CICS Monitoring is inactive.
09.36.26 JOB36923 +DFHSI1502I CICSIVP1 CICS startup is Initial.
09.36.26 JOB36923 +DFHQB0100I CICSIVP1 Temporary Storage initialization has started.
09.36.26 JOB36923 +DFHSI1503I CICSIVP1 Terminal data sets are being opened.
09.36.27 JOB36923 +DFHLG0102I CICSIVP1 Log manager domain initialization has ended.
09.36.27 JOB36923 IEC161I 080-053,DFHIVPDB,CICS CICS,DFHTEMP,,
09.36.27 JOB36923 IEC161I INST.CICSTS32.CNTL.CICS.DFHTEMP,
09.36.27 JOB36923 IEC161I INST.CICSTS32.CNTL.CICS.DFHTEMP.DATA, Chapter 32. CICS verification 355
09.36.27 JOB36923 IEC161I ICFCAT.SYSPLEX2.CATALOGB
09.36.27 JOB36923 +DFHQB0102I CICSIVP1 About to format the temporary storage data set (359 control intervals).
09.36.27 JOB36923 +DFHQB0406I CICSIVP1 825
825 CICS is about to wait for predecessors defined in the MVS automatic
```

## Note:

1 The DFHIVPDB job uses the unsuffixed SIT, DFHSIT, as used by all the CICS IVPs. It also uses some system initialization parameters included in the DFH\$SIP5 member of the SYSIN data set, to override the parameters in DFHSIT. Further, the DFH\$SIP5 member was edited to specify other system initialization parameters to create the DFHIVPDB job log shown. For information about these extra system initialization parameters used by the IVP jobs, see page “Specify system initialization parameters for the IVP jobs” on page 327.

7 If you want COBOL, C, C++ and PL/I languages, remove the comment marks from the SCEERUN and SCEERUN2 libraries, and increase the memory size for the job.

You should see messages similar to those in Figure 44 at the end of the MSGUSER section of the job output.

```
DFHLOG0302 04/24/2007 09:36:31 CICSIVP1 Journal name DFHLOG has been installed. Journal type: MVS
          CICINST.CICSIVP1.DFHLOG.
DFHLOG0302 04/24/2007 09:36:31 CICSIVP1 Journal name DFHSHUNT has been installed. Journal type: MVS
          CICINST.CICSIVP1.DFHSHUNT.
DFHLOG0744 04/24/2007 09:36:31 CICSIVP1 All records in log stream CICINST.CICSIVP1.DFHLOG have been deleted.
DFHLOG0744 04/24/2007 09:36:31 CICSIVP1 All records in log stream CICINST.CICSIVP1.DFHSHUNT have been deleted.
DFHDB8116 I 04/24/2007 09:36:32 CICSIVP1 Connection to DBCTL IM7D is proceeding. Startup Table Suffix used is IV.
DFHDB8101 I 04/24/2007 09:36:32 CICSIVP1 Connection to DBCTL IM7D is now complete. Startup Table Suffix used is IV.
DFHZC3441 I 04/24/2007 09:37:55 CICSIVP1 Orderly termination of VTAM sessions requested. ((1) Module name: DFHZSHU)
DFHRM0205 04/24/2007 09:37:58 CICSIVP1 An activity keypoint has been successfully taken.
DFHLOG0743 04/24/2007 09:37:58 CICSIVP1 Tail of log stream CICINST.CICSIVP1.DFHLOG deleted at block id
          X'00000000000000FDD'.
```

Figure 44. Sample job log output from the end of the MSGUSER section of the DFHIVPDB job

You should see messages similar to those in Figure 45 in the Printer section of the job output.

```
1DFHDB8210D Connection to DBCTL is proceeding. Check CDBC TD queue.
DFHDB8225I CICSIVP1 The DBCTL ID is IM7D. The DRA Startup Table suffix is IV.

INPUT: ASMCDDPA02MS16995-28
PART=02MS16995-28      DESC= SCREW
  AREA  INV  PROJ  DIV  UNIT  CURRENT  ON      IN      TOTAL  COUNT  BACK
        DEPT  CD    PRICE  REQMTS  ORDER  STOCK  DISBURSE  TAKEN  ORDR
1.     AA   165   11   0.152   260      0       300    4030   N      0
2.     BA   165   15   0.069   60       0       80     5000   N      0
3.     FF   554   6D   0.069   440      0       430    5000   N      0
4. 2    59   109   26   6.980   950      0      1000   5000   N      0

INPUT: ASMCDDPA02JAN1N976B
PART=02JAN1N976B      DESC= DIODE CODE-A
  AREA  INV  PROJ  DIV  UNIT  CURRENT  ON      IN      TOTAL  COUNT  BACK
        DEPT  CD    PRICE  REQMTS  ORDER  STOCK  DISBURSE  TAKEN  ORDR
1. 2    55   091   26   0.000   170     2000    170    4710   N      0
```

Figure 45. Sample job log output from the Printer section of the output of the DFHIVPDB job

## Testing the CICS-DB2 environment

This section outlines how you can test the CICS-DB2 environment. It uses Phase 5 of the DB2 installation verification procedure. It is intended as an overview of what is involved, and what you would expect to see.

To use the DB2 installation verification procedure, and Phase 5 in particular, see the DB2 V3 Administration Guide. That publication gives the latest information about the procedure, and describes the steps involved in much more detail.

### Run DB2 jobs DSNTEJ5C and DSNTEJ5P

To prepare the sample applications to be used in a CICS-DB2 environment, run the jobs DSNTEJ5C and DSNTEJ5P supplied with DB2.

Job DSNTEJ5C installs the sample application transactions in COBOL and prepares the organization application. Job DSNTEJ5P installs the transactions in PL/I and prepares the organization, project, and phone applications.

Both these jobs perform the following functions:

- Compile and link-edit the CICS online applications.
- Bind the CICS online applications.
- Create the BMS maps for the online applications.

### Starting a DB2 organization or project application

After logging on to CICS, you can start an organization or project application by entering one of the following CICS transaction codes:

- D8PP, which starts the PL/I project version
- D8PS, which starts the PL/I organization version
- D8CS, which starts the COBOL organization version

If you enter one of these transaction codes, the panels shown in Figure 46 or Figure 47 on page 358 are displayed.

```
                ACTION SELECTION
MAJOR SYSTEM ...: 0          ORGANIZATION
ACTION .....:
OBJECT .....:
SEARCH CRITERIA.:
DATA .....:
SELECT AN ACTION FROM FOLLOWING LIST

A   ADD (INSERT)
D   DISPLAY (SHOW)
E   ERASE (REMOVE)
U   UPDATE (CHANGE)
```

Figure 46. Initial panel for the DB2 project application in CICS

```
                ACTION SELECTION
MAJOR SYSTEM ...: P          PROJECTS
ACTION .....:
OBJECT .....:
SEARCH CRITERIA.:
DATA .....:
SELECT AN ACTION FROM FOLLOWING LIST

  A   ADD (INSERT)
  D   DISPLAY (SHOW)
  E   ERASE (REMOVE)
  U   UPDATE (CHANGE)
```

Figure 47. Initial panel for the DB2 project application in CICS

For detailed information about running the organization and project applications, see the DB2 V3 Administration Guide.

**Starting the DB2 phone application:** To start the phone application, clear the screen and type in the transaction code D8PT. You can change the transaction codes when you install DB2. Check with your system administrator to find out if they have been changed from those shown.

## Running the EJB "Hello World" sample

You are recommended to include running the EJB "Hello World" sample in your Installation Verification procedures.

For a description of this procedure, see *Java Applications in CICS*.

---

## **Part 6. CICSplex SM verification**

This part describes the processes and procedures you should follow to run the installation verification procedures for CICSplex SM.



## Chapter 33. CICSplex SM verification

This part describes the installation verification procedure for CICSplex SM.

### Using DFHISTAR to verify CICSplex SM has installed correctly

After you have installed CICSplex SM, you can verify that the installation has worked correctly by setting up a simple CICSplex SM environment, as shown in Figure 48, using the samples generated by running the DFHISTAR job.

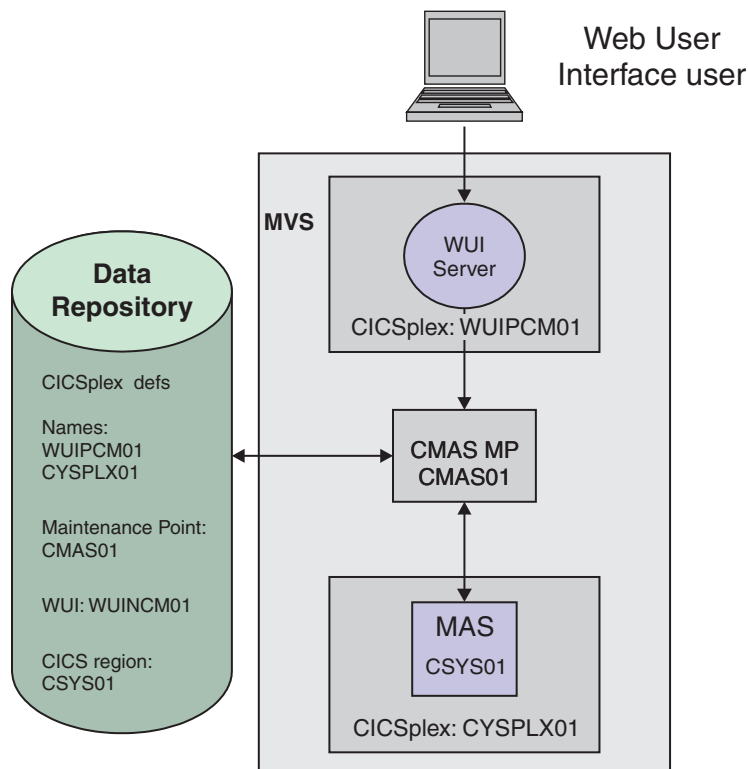


Figure 48. A simple CICSplex SM environment

CICSplex SM post-installation members are stored in the TDFHINST library. You can customize CICSplex SM post-installation members by running a tailored DFHISTAR job. Customized post-installation members are saved in the XDFHINST library. The members used to start the CMAS (EYUCMASP), WUI (EYUWUIP), and managed CICS system (EYUCSYSP) contain EYUPARM, EYUWUI, and SYSIN parameters. These parameters are held in members also stored in the XDFHINST library.

**Note:** The samples generated by running DFHISTAR have security inactive. Details of implementing RACF security for CICSplex SM are in Implementing CICSplex SM security in the *CICS RACF Security Guide*.

As part of the installation verification procedure we define a separate CICSplex for the WUI server. This is good practise for management purposes.

The steps in the installation verification procedure (IVP) for CICSplex SM are as follows:

- “Checking the MVS environment”
- “Generating samples using the DFHISTAR job”
- “Defining the data sets” on page 363
- “Creating a CICSplex SM address space (CMAS)” on page 363
- “Creating a CICSplex SM Web User Interface server” on page 364
- “Testing your Web User Interface” on page 365
- “Using the Web User Interface to define a CICSplex and a MAS” on page 366
- “Creating a CICSplex SM managed CICS system (MAS)” on page 367
- “Verifying IVP is complete using the WUI” on page 367
- “Shutting down the MAS, WUI server, and CMAS” on page 368

## Checking the MVS environment

Before you begin, check that your MVS environment is correctly defined to support CICSplex SM.

If you want to record information about the installation and set up of your CICSplex SM environment, you can use the checklists and worksheets in “CICSplex SM setup checklist and worksheets” on page 9.

1. Ensure that the CICSplex SM SEYUAUTH library is defined to MVS as an APF-authorized library.
2. Ensure that the CICSplex SM SEYULINK library is included in the MVS link list.
3. The number of linkage indexes should be increased by 1 for CICSplex SM. This is defined by the MVS NSYSLX parameter in IEASYSxx which should be increased by one.
4. The number of common MVS data spaces should be increased by adding 6 for each CMAS. This is defined by the MVS MAXCAD parameter in IEASYSxx.
5. Check the IEASYSxx member of SYS1.PARMLIB that you use for MVS initialization and make a note of the initialization values. These are referenced during the installation of a CMAS. For a detailed description of these values, see “Noting IEASYSxx values for CICSplex SM” on page 100.
6. Ensure that enough MVS auxiliary storage is available to support the data spaces used by each CMAS. A CMAS uses up to 9 MVS data spaces, which could mean an increase in auxiliary storage of as much as 100MB. For details on how to increase auxiliary storage, see the *z/OS Initialization and Tuning Guide*.
7. A VTAM APPLID and SYSID should be assigned for each of the CMAS, MAS, and Web User Interface regions. See Chapter 14, “Defining CICS regions as applications to VTAM,” on page 117. You can record these values on the planning worksheet.
8. A TCP/IP port should be reserved for the Web User Interface server. You can record the port number on the planning worksheet.

## Generating samples using the DFHISTAR job

You can tailor the DFHISTAR job to generate samples that suit your environment.



The minimum subset of parameters to tailor are as follows but you may tailor any of the parameters mentioned on the IVP planning worksheet. You may wish to record your values in the IVP planning worksheet, see “IVP planning worksheet” on page 14.

1. Edit the LIB parameter to specify where you would like the customized members generated by the DFHISTAR job to be added. You can specify a 1 to 44 character name of a library. The default is CICSTS32.XDFHINST.
2. Edit the SCOPE parameter to specify POST.
3. Edit the TCPIPHST parameter to specify the address of the host server. For example, MVSXX.COMPANY.COM.
4. Edit the TCPIPPRT parameter to specify the port number to be used by the Web User Interface server.
5. Edit the TIMEZONE parameter to specify which time zone is used for the data repository. For a full list of time zones, see *CICSplex SM Administration*.
6. Run the tailored DFHISTAR job.

**Note:**

- DFHISTAR creates the high level qualifiers for the CICS and CICSplex SM load libraries using 2 parameters: TINDEXT provides the highest level index and XTRAQUAL is an optional lower level index. The index ‘CICS’ or ‘CPSM’ appears after the TINDEXT and before the XTRAQUAL.
- DFHISTAR uses the same parameter for the system name and the applid. This is a recommended convention.

See “CICSplex SM post-installation members” on page 217 for a complete list of samples generated by running DFHISTAR.

## Defining the data sets

The samples generated by DFHISTAR include members to create the CICS and CICSplex SM data sets for a CMAS, WUI server, and MAS. Run the following jobs:

1. EYUCMSDS. This creates the CICS and CICSplex SM data sets for a CMAS. By default, DFHISTAR customizes the DREPINIT job step, within EYUCMSDS, to create a CMAS data repository and add definitions for the CMAS, a WUI server, and a CICSplex for that WUI server.
2. EYUWUIDS. This creates the CICS and CICSplex SM data sets for a WUI server.
3. EYUCSYDS. This creates the CICS and CICSplex SM data sets for a MAS.

## Creating a CICSplex SM address space (CMAS)

The sample member, EYUCMS0P, generated by DFHISTAR contains customized data for the following CICSplex SM parameter:

NAME(CMAS01)            CMAS Name (Default is APPLID)

For a complete list of CICSplex SM parameters, see “CICSplex SM system parameters” on page 305.

To create a CMAS:

1. Review and edit the CMAS-related CICS system initialization table (SIT) parameters in sample member EYUCMSSP. You should edit the CICS SVC number (CICSSVC) if you use a number other than the CICS default SVC number.
2. Run the sample member EYUCMASJ to start the CMAS.

The CICS region will start and the CMAS will initialize. Check the job log for message

```
EYUXL0010I applid CMAS initialization complete
```

## Creating a CICSplex SM Web User Interface server

A WUI server region runs as a MAS and is managed by the CMAS. Its operation is controlled by the CICSplex SM parameters that are required for a MAS and the WUI server initialization parameters.

The sample member, EYUWUIOP, generated by DFHISTAR contains customized data for the following CICSplex SM parameters.

```
NAME(WUINCM01)      WUI Name (Default is APPLID)
CICSplex(WUIPCM01)  CICSplex to which the WUI connects
CMASYSID(CM01)     CMAS to which the WUI connects
```

For a complete list of CICSplex SM parameters, see “CICSplex SM system parameters” on page 305.

The sample member, EYUWUIIN, generated by DFHISTAR contains customized data for the following WUI server initialization parameters.

```
TCPIPHOSTNAME(@tcpiphst@) TCP/IP host name of this WUI Server
TCPIPPORT(@tcpipprt@)     TCP/IP port number
DEFAULTCMASCTX(@cmasname@) CMAS context - CMAS name
DEFAULTCONTEXT(@wuiplex@) Context - CICSplex name
DEFAULTSCOPE(@wuiplex@)   Scope - CICSplex, CICS group or MAS name
*****
* AUTOIMPORTDSN is required only when importing view sets from
* a data set. Use it when starting the WUI for the first time
* or when importing new or modified view sets.
*
* The SEYUVIEW data set contains the IBM-supplied menus and view
* sets.
*
* To import menus and view sets for languages other than English,
* change AUTOIMPORTMEM to:
*
* EYUSA* for Chinese
* EYUKA* for Japanese
*
*****
AUTOIMPORTDSN(@thlq@.CPSM.@tqual@.SEYUVIEW)
AUTOIMPORTMEM(EYUEA*)      Import the English menus and view sets
*****
```

For a complete list of Web User Interface server initialization parameters, see “Specify the Web User Interface server initialization parameters” on page 292.

To create a CICSplex SM Web User Interface server:

1. Review and edit the WUI-related CICS system initialization table (SIT) parameters in sample member EYUWUIISP.

- You should edit the CICS SVC number (CICSSVC) if you use a number other than the CICS default SVC number.
- The optional INITPARM parameter is set to the default of English. INITPARM sets the language and codepage for the WUI. If you want to use another language specify the value for INITPARM using the format:

```
INITPARM=(EYU9VKEC='xxx',EYU9VWAN='yyyy')
```

where xxx represents the language of the Web User Interface server, and yyyy represents the codepage of the client. For example, if your chosen language is Japanese, code:

```
INITPARM=(EYU9VKEC='JPN',EYU9VWAN='JPN1').
```

For a list of language and codepage identifiers, see “Specify language and codepage” on page 287.

2. Prepare the codepage conversion table DFHCNV. For information on how to do this see “Prepare codepage conversion table” on page 288
3. Run the sample member EYUWUIJ to start the WUI.

This will automatically import the Web User Interface views into the server repository. Check the EYULOG for messages:

```
EYUVS0002I CICSPlex SM Web User Interface initialization complete.
```

```
EYUVS0010I Server connected to CMAS, SYSID(sysid).
```

## Testing your Web User Interface

Once the WUI views have been imported and WUI initialization has completed you can test that your Web User Interface is running as follows:

1. Enter the following URL: `http://hostname:portnumber` where
  - hostname is the TCP/IP host name on the TCPIPHOSTNAME parameter in the EYUWIIN member (specified as TCPIPHST for DFHISTAR)
  - portnumber is the TCP/IP port number on the TCPIPPORT parameter in the EYUWUIIN member (specified as TCPIPPRT for DFHISTAR).

After you have signed into your system you will see the main menu for the Web User Interface as shown in Figure 49 on page 366

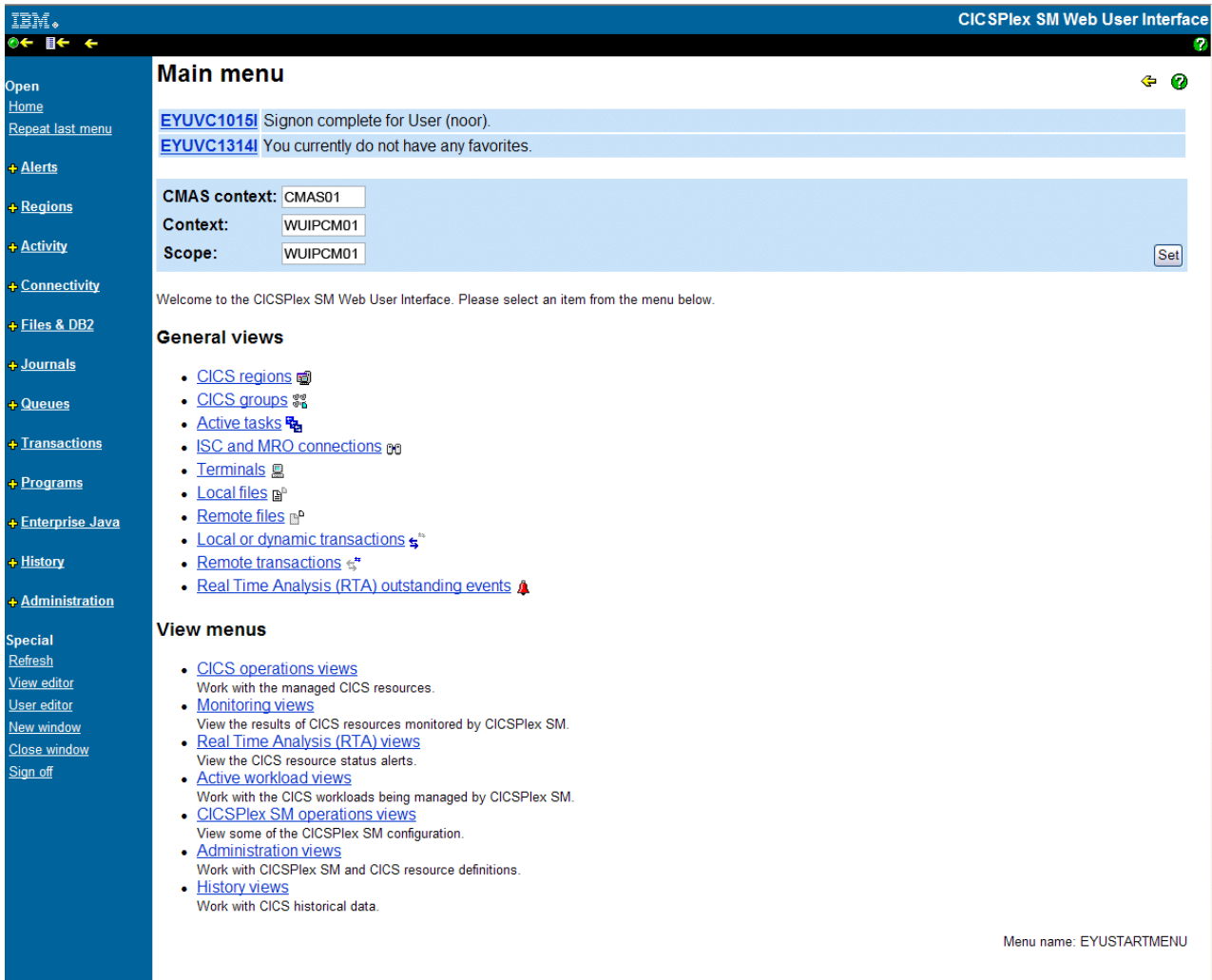


Figure 49. The Web User Interface main menu

2. From the main menu, select the CICS regions option. Details of the WUI server region are displayed. By default, DFHISTAR customizes EYUWUJIN to set the DEFAULTCONTEXT and DEFAULTSCOPE to the name of the CICSPlex defined for the WUI.

## Using the Web User Interface to define a CICSPlex and a MAS

Before starting the MAS use the WUI to define a CICSPlex and a MAS as follows:

1. Define a CICSPlex for your MAS, starting from the WUI main menu, as follows:
  - Click Administration views > CMAS configuration administration views > CICSPlex definitions
  - Create a CICSPlex definition:
    - Click the Create button
    - Specify CICSPlex using the name specified in the CICSPLEX parameter in member EYULMSOP. You can leave the default values in the other fields.
    - Click the Yes button
2. Add a MAS to your newly defined CICSPlex, starting from the WUI main menu, as follows:

- Change the Context and Scope fields to the name of the CICSplex and click on the Set button. The name should match the CICSPLEX parameter in the EYULMS0P member.
- Click CICSplex SM operations views > CICS system definitions.
- Click on the Create button to create a CICS system definition.
- Set the CICS system definition name using the NAME parameter from EYULMS0P.
- Set the Primary CMAS name using the NAME parameter from EYUCMS0P.
- You can leave the Period definition name blank. A PERIODEF is needed to set time periods for the CICSplex SM system availability resource monitoring (SAM). We are not activating SAM in this IVP.
- Set the MAS Application ID using the APPLID parameter from EYULMSSP.
- Set the MAS System ID using the SYSIDNT parameter from EYULMSSP.

## Creating a CICSplex SM managed CICS system (MAS)

The sample member, EYULMS0P, generated by DFHISTAR contains customized data for the following CICSplex SM parameters:

NAME(CSYS01)	MAS Name (Default is APPLID)
CICSPLEX(CSYPLX01)	CICSplex to which the MAS is associated with
CMASYSID(CM01)	CMAS to which the MAS connects

For a complete list of CICSplex SM parameters, see “CICSplex SM system parameters” on page 305.

To create a MAS:

1. Review and edit the MAS-related CICS system initialization table (SIT) parameters in sample member EYULMSSP. You should edit the CICS SVC number (CICSSVC) if you use a number other than the CICS default SVC number.
2. Run the sample member EYUCSYSJ to start the MAS.

The CICS region will start and the MAS will initialize. Check the MAS job log for message

```
EYUNL0099I LMAS LRT initialization complete.
```

```
EYUTS0003I Topology event for sysname Complete - APPLID (applid) CICSplex (plexname)
```

Check the CMAS job log for message

```
EYUTS0003I Topology event for sysname Complete - APPLID (applid) CICSplex (plexname)
```

## Verifying IVP is complete using the WUI

To verify that the IVP is complete:

1. Click CICS regions from the WUI main menu. Ensure the Context and Scope fields are set to the CICSplex name used as the CICSPLEX parameter in EYULMS0P. Details of your managed CICS system are displayed.
2. Click Administration views > CMAS configuration administration > CICSplex definitions Two CICSplexes are shown, one created by the EYU9XDUT step in the EYUCMSDS job and the other by you using the WUI.

The IVP is now complete.

## Shutting down the MAS, WUI server, and CMAS

Shutdown the systems as follows:

1. To shutdown the MAS see “Terminating a MAS” on page 284.
2. To shutdown the WUI server see “Shutdown the Web User Interface server” on page 304.
3. To shutdown the CMAS see “Shutting down a CMAS” on page 273.

---

## Setting up a more complex configuration

You can continue to define more managed CICS systems and add them to your CICSplex. In a larger configuration, you might have several CMASs. A CMAS that is specified as the context when you create a CICSplex definition is the maintenance point CMAS for that CICSplex. When applying service to CICSplex SM, some PTFs must be applied to the maintenance points first, before the other regions are updated. For this reason, it is beneficial to keep the number of maintenance point CMASs to a minimum.

To increase availability, it is recommended that you have several WUIs, but each CMAS does not require a unique WUI. Connecting a WUI to a CMAS allows you to link the CMAS to others in your network. After building your CMAS to CMAS links, you can remove the WUIs and their CICSplex definitions from some of your CMASs to reduce the number of maintenance point CMASs.

---

## Part 7. Appendixes





---

## Appendix A. Information about modules eligible for the MVS link pack area

The following information is provided in Table 27 on page 375 and Table 28 on page 375. Some of the information applies only to the modules listed in Table 28 on page 375.

**Name** The name of the module.

**Description**

A brief description of the module. This gives some clues to the associated function, useful if the module does not have a controlling CICS option.

**Library**

(Table 28 on page 375 only.) The library in which the module is installed:

**Library**

**DS name**

**AUTH** *hlq.SDFHAUTH*

**LOAD** *hlq.SDFHLOAD*

**LINK** *SYS1.hlq.SDFHLINK*

You can use the CICS-supplied usermods to move the modules from these libraries to the *hlq.SDFHLPA* library.

**LPA/ELPA**

(Table 28 on page 375 only.) In this column, the terms **LPA** and **ELPA** are used to indicate whether a module is loaded into the part of the MVS link pack area that is below (LPA) or above (ELPA) 16MB.

**Priority**

(Table 28 on page 375 only.) A nominal “priority” to help you decide whether a module should be in the MVS link pack area and to choose between modules if your MVS link pack area is short on space.

**Size** The size of the module.

**Option/Note**

Identifies one or more notes about the use of the module from the MVS link pack area and any associated CICS options to be specified for the function that uses the module.

Some of these information categories are described in more detail in the following sections.

---

### Priority

The priority of the modules eligible for the LPA are as follows:

1. Must be in the MVS link pack area. Information about these modules, installed in the *hlq.SDFHLPA* library, is given in Table 27 on page 375.
2. Generally a good candidate for inclusion in the MVS link pack area. You should include these modules in the LPA to support the associated option.
3. A good candidate for inclusion in the MVS link pack area. You should include these modules in the MVS link pack area if you are a heavy user of the associated function.

---

## Size

The module sizes were taken from the latest information available at the time of publishing, but may be different in your CICS environment depending on the options selected and if any PTFs applied affect the modules. The sizes are given here to help you plan the amount of storage that you need for the modules that you want to install in the MVS link pack area. You can get the actual sizes for these modules from a directory listing of the modules or from the module index provided at the back of a formatted SDUMP taken with the LPA=NO system initialization parameter specified.

---

## Option/Note

This column identifies any CICS options associated with the use of the module from the MVS link pack area, or refers to a note in the following list for additional information, or both.

### Note:

1. The program is used from the MVS link pack area only if you set the USELPACOPY option of its program resource definition to YES.
2. You must always install the latest service level of the CICS SVC module, DFHCSVC. You should install the DFHCSVC module into the MVS link pack area before you run the CICS installation verification procedures.

You must define the DFHCSVC module in an IEASVCxx member of the SYS1.PARMLIB library, using SVC Parm statements. You select the required IEASVCxx member by coding the SVC parameter (SVC=xx) in a SYS1.PARMLIB member (IEASYSyy), which you use to IPL your MVS.

You can run several CICS regions, at different release levels, in the same MVS image, with each region using its own version of the DFHCSVC module. However, if some of those regions use MRO, then all regions that use MRO must use the latest DFHCSVC module and the latest DFHIRP module.

If you have some regions that are to use the DFHCSVC module, and you give the SVC a number different from the SVC number used by the regions, you must generate a new version of the DFHCRC program on the regions.

For information about defining and using the DFHCSVC module, see the *CICS Transaction Server for z/OS Program Directory*.

3. If your batch region is sharing the database with a CICS/OS/VS 1.7 region or a CICS/MVS Version 2 region, you can continue to use the batch region controller program, DFHDRP, from before CICS/ESA Version 3. (The CICS/ESA Version 3 DFHIRP program supports earlier levels of the DFHDRP program.) However, if your batch region is sharing the database with a CICS TS for z/OS, Version 3.2 region, you are recommended to install the CICS TS for z/OS, Version 3.2 DFHDRP module in SYS1.LINKLIB, or another suitable APF-authorized library in the MVS linklist.
4. All LPA-required modules are downward-compatible with earlier releases of CICS. If you are running earlier releases of CICS, you must ensure that the correct version is installed in the LPA. The module must be in the LPA for integrity reasons, but the post exit routine itself can reside either in the LPA, or in the CICS address space. This enables

you to use different versions of the DFHDSAUT module in different CICS regions running in the same MVS image, because the DFHDSAUT module may not be compatible from release to release.

5. The use of this pre-CICS/ESA Version 3 programmable interface to the master terminal program, DFHEMTA, is supported for compatibility reasons only. You are strongly recommended to use the equivalent EXEC CICS INQUIRE and ISET commands instead. The documentation for this interface is available only in the CICS libraries for the releases prior to CICS/ESA Version 3.
6. You can set the system tracing status by coding appropriate system initialization parameters, and you can also set it dynamically by using the CETR transaction.

The system initialization parameters that you can use are:

<b>Parameter</b>	<b>Use</b>
------------------	------------

<b>AUXTR</b>	Activate auxiliary trace.
--------------	---------------------------

<b>AUXTRSW</b>	Define the auxiliary switch status.
----------------	-------------------------------------

<b>GTFTR</b>	Enable CICS to use MVS GTF tracing.
--------------	-------------------------------------

<b>INTTR</b>	Activate CICS internal tracing.
--------------	---------------------------------

<b>TRTABSZ</b>	Specify the size of the internal trace table.
----------------	---

<b>USERTR</b>	Set the master user trace flag on or off.
---------------	---

For information about using CICS trace, and using the CETR transaction to control the tracing status, see *CICS Problem Determination Guide*.

7. The DFHIRP module needs to be in the MVS link pack area only if you are using MRO, CICS shared database, or the console message-handling facility. If you install the DFHIRP module in the MVS link pack area, you must also install DFHSEN if you are using the console message-handling facility.  
You must always install the latest service level of the DFHIRP (if needed) and DFHSEN.  
If you are running CICS with MRO at different release levels, all regions in the same MVS-image must use the latest DFHIRP module.
8. To use the console message formatting facility of the MVS subsystem interface, you must install the modules DFHSSGC and DFHSSWT either in the MVS link pack area or in an APF-authorized library in the MVS linklist. These modules are used by the subsystem interface and not directly by CICS. Therefore, the use of these modules from the MVS link pack area is not controlled by CICS parameters or options.  
For information about enabling the console message-formatting facility, and about the other modules it requires, see “Modules needed to use the console message-handling facilities” on page 107.
9. CICS needs the following load modules, supplied with CICS, to use data table services:
  - DFHDTINT

- DFHDTOC
- DFHDTLD
- DFHDTRD
- DFHDTES

The modules are all eligible for the MVS link pack area, but DFHDTRD and DFHDTES are probably the only ones which are used sufficiently frequently to be worth considering.

10. The following modules, used by the Shared Data Tables facility, are eligible for the MVS link pack area:

- DFHDTAM
- DFHDTAOR
- DFHDTCV
- DFHDTFOR
- DFHDT SVC
- DFHDTXS
- DFHMVRMS

All these modules, except for DFHMVRMS, are listed in the usermod, DFH\$UMOD, supplied with CICS. Only DFHDTAM, DFHDTAOR, DFHDTFOR, and possibly DFHDTCV are used sufficiently frequently to be worth considering for the MVS link pack area.

The following modules, installed in the *hlq.SDFHLINK* library, must be installed in the MVS linklist, or in the MVS link pack area:

- DFHDT SVC
- DFHDTCV
- DFHMVRMS

11. BMS=STANDARD
12. BMS=FULL
13. BMS=MINIMUM
14. DTRPGM=DFHDYP
15. SPOOL=YES
16. FCT=YES/xx
17. ISC=YES/xx
18. VTAM=YES
19. XRF=YES/xx
20. AUXTR=ON
21. TST=YES/xx
22. TCP=YES/xx

This column also gives any associated options that you must specify to use the function associated with the LPA-eligible module. Unless otherwise stated, the options are specified by system initialization parameters as defined in *CICS System Definition Guide*

Table 27. LPA-required modules, supplied in hlq.SDFHLPA

Name	Description	LPA/ ELPA	Size	Option/ Note
DFHC SVC	CICS SVC startup	ELPA	2280	CICSSVC (2 on page 372) (4 on page 372)
DFHDSPEX	DS domain - MVS POST exit stub	ELPA	168	(4 on page 372)
DFHDUMPX	SDUMPX IEASDUMP QUERY exit	ELPA	152	(4 on page 372)
DFHIRP	Interregion communication program	ELPA	49416	(7 on page 373) (4 on page 372)
DFHSEN	Subsystem interface end-of-memory / end-of-task clean up routine	ELPA	472	(4 on page 372)
DFHSSGC	Subsystem interface generic connect	ELPA	936	(8 on page 373) (4 on page 372)
DFHSSWT	Subsystem interface WTO router	ELPA	4512	(8 on page 373) (4 on page 372)
DFH99SVC	Dyn alloc - SVC services	ELPA	8	(4 on page 372)

Table 28. LPA-eligible modules

Name	Description	Library	LPA/ ELPA	Pri	Size	Option/ Note
AXMSC	Server connection routines	LINK	ELPA	2	21856	-
EYUMCT1C		AUTH	ELPA	2		-
EYUMCT1E		AUTH	ELPA	2		-
EYUMCT1K		AUTH	ELPA	2		-
DFHADWM0		LOAD	ELPA	2		-
DFHADWM1		LOAD	ELPA	2		-
DFHADWM2		LOAD	ELPA	2		-
DFHADWM3		LOAD	ELPA	2		-
DFHADWM4		LOAD	ELPA	2		-
DFHADWM5		LOAD	ELPA	2		-
DFHADWM6		LOAD	ELPA	2		-
DFHADWT0		LOAD	ELPA	2		-
DFHADWT1		LOAD	ELPA	2		-
DFHADWT2		LOAD	ELPA	2		-
DFHADWT3		LOAD	ELPA	2		-
DFHADWT4		LOAD	ELPA	2		-
DFHADWT5		LOAD	ELPA	2		-
DFHAIIN	AITM Manager initialization	LOAD	ELPA	3	2048	AIEXIT
DFHAIIQ	AITMM - locate/unlock/inquire/browse	LOAD	ELPA	2	1384	AIEXIT
DFHAIP	Application Interface program	LOAD	LPA	2	11560	-
DFHAIRP	AITMM - initialization/recovery	LOAD	ELPA	3	1592	-
DFHAITM	AITMM - add replace/delete	LOAD	ELPA	3	3216	AIEXIT
DFHALP	Terminal allocation	LOAD	ELPA	2	21784	AIEXIT

Table 28. LPA-eligible modules (continued)

Name	Description	Library	LPA/ ELPA	Pri	Size	Option/ Note
DFHALRC		LOAD	ELPA	2		-
DFHALXM		LOAD	ELPA	2		-
DFHAPAC	AP domain - abnormal condition reporting interface module	LOAD	ELPA	3	1928	-
DFHAPATT	AP domain - entrypoint attach	LOAD	ELPA	2	728	-
DFHAPCR		LOAD	ELPA	3		-
DFHAPDM	AP domain - initialization/termination	LOAD	ELPA	3	5416	-
DFHAPDN	AP domain - transaction definition notify	LOAD	ELPA	3	2904	-
DFHAPEP	AP domain - user exit service	LOAD	ELPA	2	11128	-
DFHAPID		LOAD	ELPA	3		-
DFHAPIN	AP domain - special initialization for programs and user-replaceable modules	LOAD	ELPA	2	184	-
DFHAPIQ	AP domain - user exit data access service	LOAD	ELPA	3	1232	-
DFHAPJC	AP domain - journaling gate service	LOAD	ELPA	3	2528	-
DFHAPLI	AP domain - language interface program	LOAD	ELPA	2	27528	-
DFHAPLJ		LOAD	ELPA	2		-
DFHAPLX	AP domain - manage execution under X8/X9 TCB	LOAD	ELPA	2		-
DFHAPNT	AP domain - MXT notify gate	LOAD	ELPA	3	1096	-
DFHAPPG	AP domain - optimize initial_link for	LOAD	ELPA	2	1808	-
DFHAPRDR	AP domain gate APRD	LOAD	ELPA	2	22176	-
DFHAPRT	AP Domain - route transaction gate	LOAD	ELPA	3	9104	-
DFHAPSTL	AP domain - statistics collection program	LOAD	ELPA	2	35248	-
DFHAPTC		LOAD	ELPA	3		-
DFHAPTI	AP domain - timer notify gate	LOAD	ELPA	2	1096	-
DFHAPTIX	AP domain - expiry analysis task	LOAD	ELPA	2	1096	-
DFHAPXM	AP domain - transaction initialization and termination services	LOAD	LPA	2	3744	-
DFHAPXME	AP domain - XM exception handler	LOAD	ELPA	3	2720	-
DFHASV	Authorized services interface	AUTH	LPA	2	2504	-
DFHBADML		LOAD	ELPA	2		-
DFHBRAI	Bridge Functions	LOAD	ELPA	3	??	-
DFHBRAT	Bridge BRAT gate Functions	LOAD	ELPA	3	AA0	-
DFHBRFM	Bridge Facility Management	LOAD	ELPA	3	3620	-
DFHBRFR	Bridge Functions	LOAD	ELPA	3	??	-
DFHBRIC	IC Bridge Exit Interface	LOAD	ELPA	2	1570	-
DFHBRIQ	Bridge BRIQ gate Functions	LOAD	ELPA	3	580	-
DFHBRLK	Bridge Functions	LOAD	ELPA	3	??	-
DFHBRME	Bridge Functions	LOAD	ELPA	3	??	-
DFHBRMF	Bridge Functions	LOAD	ELPA	3	??	-
DFHBRMG	Bridge Functions	LOAD	ELPA	3	??	-
DFHBRMR	Bridge Functions	LOAD	ELPA	3	??	-

Table 28. LPA-eligible modules (continued)

Name	Description	Library			Pri	Size	Option/ Note
			LPA/ ELPA				
DFHBRMS	BMS Bridge Exit Interface	LOAD	ELPA		2	49B0	-
DFHBRNS	Bridge Functions	LOAD	ELPA		3	??	-
DFHBRRM	Bridge Recovery Manager Functions	LOAD	ELPA		3	8D0	-
DFHBRSP	SP Bridge Exit Interface	LOAD	ELPA		2	D60	-
DFHBRTB	Bridge terminal buffer management program	LOAD	ELPA		2	3F90	-
DFHBRTC	Bridge support for Terminal Control requests	LOAD	ELPA		2	3D08	-
DFHBRTQ	Bridge terminal buffer query program	LOAD	ELPA		2	2AC0	-
DFHBRXM	Bridge XM Attach Client	LOAD	ELPA		3	21E0	-
DFHCCNV	Data conversion for CICS OS/2 ISC users	LOAD	ELPA		2	806096	(1 on page 372)
DFHCCNVG		LOAD	ELPA		3		-
DFHCDKRN		LOAD	ELPA		3		-
DFHCEGN	Goodnight transaction stub	LOAD	ELPA		3	2400	(1 on page 372)
DFHCESD	Shutdown transaction	LOAD	ELPA		3	4088	(1 on page 372)
DFHCHS	CICS/ESA mirror for CICS OS/2 and CICS/VM	LOAD	ELPA		2	8368	(1 on page 372)
DFHCMAC	ME domain - CICS messages and codes transaction (CMAC)	LOAD	ELPA		2	7576	(1 on page 372)
DFHCMP	CICS monitoring compatibility interface	LOAD	ELPA		2	496	-
DFHCNV		LOAD	ELPA		3		-
DFHCPIC	SAA communications interface program	LOAD	ELPA		2	175952	-
DFHCPIN	CPI initialization program	LOAD	ELPA		3	2744	-
DFHCPIRR	SAA resource recovery interface program	LOAD	ELPA		2	1176	-
DFHCPSM	Translator for CICSplex SM commands	LOAD	LPA		3	3184	-
DFHCQCQ		LOAD	ELPA		3		-
DFHCRLB	Subroutine entry to process logging requests for MRO and LU6.1 links from assembler callers	LOAD	ELPA		2	1360	-
DFHCRNP	Interregion connection manager	LOAD	ELPA		2	10696	(1 on page 372)
DFHCRQ	ATI purge program	LOAD	ELPA		2	872	(1 on page 372)
DFHCRR	Interregion session recovery program	LOAD	ELPA		3	4840	(1 on page 372)
DFHCRS	Remote scheduler program	LOAD	ELPA		2	6800	(1 on page 372)
DFHCRSP	CICS IRC startup module	LOAD	ELPA		3	3528	(1 on page 372)
DFHCRT	Transaction routing relay program for APPC devices	LOAD	ELPA		2	648	(1 on page 372)

Table 28. LPA-eligible modules (continued)

Name	Description	Library	LPA/ ELPA	Pri	Size	Option/ Note
DFHCRU	Performs initialization of AP gates for various resource managers and facilities	LOAD	ELPA	2	34448	-
DFHDBAT	CICS-DBCTL adapter/transformer	LOAD	ELPA	2	7960	(1 on page 372)
DFHDBCT	CICS-DBCTL control program	LOAD	ELPA	3	16968	(1 on page 372)
DFHDBMOX	CICS-DBCTL monitoring exit	LOAD	ELPA	2	488	-
DFHDBREX	CICS-DBCTL resume exit	LOAD	ELPA	2	472	-
DFHDBSPX	CICS-DBCTL suspend exit	LOAD	ELPA	2	1000	-
DFHDBSSX	CICS-DBCTL status exit	LOAD	ELPA	3	672	-
DFHDBSTX	CICS-DBCTL statistics exit	LOAD	ELPA	3	656	-
DFHDBTI	EXEC DLI LD table	LOAD	ELPA	3	9024	(1 on page 372)
DFHDBTOX	CICS-DBCTL token exit	LOAD	ELPA	3	488	-
DFHDCP	Dump control program	LOAD	ELPA	3	848	-
DFHDHML		LOAD	ELPA	2		-
DFHDHRP		LOAD	ELPA	3		-
DFHDIP	Data interchange program	LOAD	ELPA	2	3936	DIP=YES
DFHDIPDY	Data interchange program (dummy)	LOAD	ELPA	2	168	DIP=NO
DFHDLI	DL/I call router	LOAD	LPA	2	4504	-
DFHDLIDP	DBCTL call processor	LOAD	ELPA	2	7032	-
DFHDLLOD		LOAD	ELPA	2		-
DFHDMRM	CSD open/close program	LOAD	ELPA	3	888	-
DFHDPML		LOAD	ELPA	3		-
DFHDPWM0		LOAD	ELPA	2		-
DFHDPWM1		LOAD	ELPA	2		-
DFHDPWM2		LOAD	ELPA	2		-
DFHDPWM3		LOAD	ELPA	2		-
DFHDPWM4		LOAD	ELPA	2		-
DFHDPWM5		LOAD	ELPA	2		-
DFHDPWM6		LOAD	ELPA	2		-
DFHDPWT0		LOAD	ELPA	2		-
DFHDPWT1		LOAD	ELPA	2		-
DFHDPWT2		LOAD	ELPA	2		-
DFHDPWT3		LOAD	ELPA	2		-
DFHDPWT4		LOAD	ELPA	2		-
DFHDPWT5		LOAD	ELPA	2		-
DFHDPWT6		LOAD	ELPA	2		-
DFHDSAUT	DS domain - authorized services	AUTH	ELPA	2	2280	-
DFHDSBA\$	BMS data stream build (standard)	LOAD	ELPA	2	1592	(11 on page 374)
DFHDSB1\$	BMS data stream build (full)	LOAD	ELPA	2	1592	(12 on page 374)
DFHDSRP		LOAD	ELPA	2		-
DFHDTAM	Shared data tables: access manager	AUTH	ELPA	2	11680	(10 on page 374)
DFHDTAOR	Shared data tables: AOR module	AUTH	ELPA	2	3264	(10 on page 374)



Table 28. LPA-eligible modules (continued)

Name	Description	Library	LPA/ ELPA	Pri	Size	Option/ Note
DFHDTCV	Shared data tables connection validation	LINK	ELPA	2	288	(10 on page 374)
DFHDTFOR	Shared data tables: FOR module	AUTH	ELPA	2	13400	(10 on page 374)
DFHDTLX		LOAD	ELPA	2		-
DFHDTSVC	Shared data tables: SVC services	LINK	ELPA	3	10944	(10 on page 374)
DFHDTXS	Shared data tables connection security	AUTH	ELPA	3	1616	(10 on page 374)
DFHDUIO	DU domain - open/close/switch/write	LOAD	LPA	2	5016	-
DFHDUSVC	DU domain - SVC processing routine	AUTH	ELPA	2	4160	-
DFHDYP	Dynamic routing program	LOAD	ELPA	2	320	(14 on page 374) (1 on page 372)
DFHD2CC		LOAD	ELPA	3		-
DFHD2CM0		LOAD	ELPA	3		-
DFHD2CM1		LOAD	ELPA	3		-
DFHD2CM2		LOAD	ELPA	3		-
DFHD2CM3		LOAD	ELPA	3		-
DFHD2CO		LOAD	ELPA	3		-
DFHD2D2		LOAD	ELPA	3		-
DFHD2EDF		LOAD	ELPA	3		-
DFHD2EX1		LOAD	ELPA	3		-
DFHD2EX2		LOAD	ELPA	3		-
DFHD2EX3		AUTH	ELPA	3		-
DFHD2IN		LOAD	ELPA	3		-
DFHD2INI		LOAD	ELPA	3		-
DFHD2MSB		AUTH	ELPA	3		-
DFHD2RP		LOAD	ELPA	3		-
DFHD2STP		LOAD	ELPA	3		-
DFHD2STR		LOAD	ELPA	3		-
DFHD2TM		LOAD	ELPA	3		-
DFHEBU	EXEC FMH construction	LOAD	ELPA	2	432	(17 on page 374)
DFHECBAM		LOAD	ELPA	3		-
DFHECID	CECI service program	LOAD	ELPA	3	77424	(1 on page 372)
DFHECIP	Command interpreter (CECI) program	LOAD	ELPA	3	3008	(1 on page 372)
DFHECSP	Command syntax check (CECS) program	LOAD	ELPA	3	3008	(1 on page 372)
DFHEDAD	RDO (CEDA) service program	LOAD	ELPA	3	127688	(1 on page 372)
DFHEDAP	RDO (CEDA) program	LOAD	ELPA	3	3144	(1 on page 372)
DFHEDC	EXEC interface for dump control	LOAD	ELPA	2	152	-
DFHEDCP	EXEC interface for dump system/transaction	LOAD	ELPA	3	3776	-

Table 28. LPA-eligible modules (continued)

Name	Description	Library	LPA/ ELPA	Pri	Size	Option/ Note
DFHEDFBR	Temporary-storage browse transaction, CEBR	LOAD	ELPA	3	12024	(1 on page 372)
DFHEDFD	EDF display program	LOAD	ELPA	3	63552	(1 on page 372)
DFHEDFE	EDF attach error handler	LOAD	ELPA	3	1360	(1 on page 372)
DFHEDFP	EDF control program	LOAD	ELPA	3	7512	(1 on page 372)
DFHEDFR	EDF response table	LOAD	ELPA	3	576	(1 on page 372)
DFHEDFX	EDF task switch program	LOAD	ELPA	3	4296	(1 on page 372)
DFHEDI	EXEC interface for data interchange	LOAD	ELPA	2	1352	DIP=YES
DFHEDP	EXEC DLI command stub	LOAD	ELPA	2	7128	(1 on page 372)
DFHEEI	EXEC interface for HANDLE, ADDRESS, ASSIGN	LOAD	ELPA	2	6536	-
DFHEEX	EXEC FMH extraction	LOAD	ELPA	2	752	-
DFHEGL	EXEC interface for unmapped LU6.2 commands	LOAD	ELPA	2	3664	(18 on page 374)
DFHEIACQ	EXEC ACQUIRE TERMINAL	LOAD	ELPA	3	1464	-
DFHEIBAM		LOAD	ELPA	2		-
DFHEICRE	EXEC CICS CREATE commands	LOAD	ELPA	3	86880	-
DFHEIDEF		LOAD	ELPA	3		-
DFHEIDEL		LOAD	ELPA	3		-
DFHEIDH		LOAD	ELPA	2		-
DFHEIDLI	DL/I load table	LOAD	LPA	3	9248	DLI
DFHEIDTI	EXEC ask-time, format-time program	LOAD	ELPA	2	3048	-
DFHEIEM		LOAD	ELPA	2		-
DFHEIFC	File control: EXEC interface module	LOAD	ELPA	2	11800	-
DFHEIGDS	Translator table (GDS commands)	LOAD	ELPA	3	2752	(1 on page 372)
DFHEIGDX	EXEC interface load table	LOAD	LPA	3	3024	-
DFHEIIC	EXEC interface IC module	LOAD	ELPA	2	7824	-
DFHEIINS		LOAD	ELPA	3		-
DFHEIPI		LOAD	ELPA	2		-
DFHEIPRT	EXEC interface for perform resettime	LOAD	ELPA	3	680	-
DFHEIPSE	EXEC interface for perform security	LOAD	ELPA	3	896	SEC=YES
DFHEIPSH	EXEC interface for perform shutdown	LOAD	ELPA	3	2800	-
DFHEIQBA		LOAD	ELPA	3		-
DFHEIQBR	EXEC bridge inquire facility	LOAD	ELPA	3	5848	-
DFHEIQCF		LOAD	ELPA	3		-
DFHEIQDH		LOAD	ELPA	3		-
DFHEIQDI		LOAD	ELPA	3		-
DFHEIQDN	EXEC inquire/set for external data sets	LOAD	ELPA	3	7576	-

Table 28. LPA-eligible modules (continued)

Name	Description	Library	LPA/ ELPA	Pri	Size	Option/ Note
DFHEIQDS	EXEC inquire/set/discard for files	LOAD	ELPA	3	14344	-
DFHEIQDU	EXEC inquire/set for dump data sets and dump codes	LOAD	ELPA	3	8584	-
DFHEIQD2		LOAD	ELPA	3		-
I DFHEIQEJ	EXEC inquire EJB commands	LOAD	ELPA	3	12824	-
DFHEIQIR	EXEC inquire/set for IRC	LOAD	ELPA	3	1976	-
DFHEIQIS		LOAD	ELPA	3		-
DFHEIQMS	EXEC inquire/set for monitor and stats	LOAD	ELPA	3	13944	-
DFHEIQMT	EXEC inquire/set for CEMT-only commands	LOAD	ELPA	3	3408	-
I DFHEIQOP	EXEC inquire requestmodel	LOAD	ELPA	3	2608	-
DFHEIQPI		LOAD	ELPA	3		-
DFHEIQRQ	EXEC inquire for queued requests (REQIDs)	LOAD	ELPA	3	3152	-
DFHEIQR		LOAD	ELPA	3		-
DFHEIQSA	EXEC inquire/set for system attributes	LOAD	ELPA	3	8928	-
DFHEIQSC	EXEC inquire/set for connections	LOAD	ELPA	3	16616	-
DFHEIQSJ	EXEC inquire/set for journals	LOAD	ELPA	3	3480	-
DFHEIQSK	EXEC inquire/set for tasks	LOAD	ELPA	3	15248	-
DFHEIQSL	EXEC inquire/discard for journalmodel	LOAD	ELPA	3	3144	-
DFHEIQSM	EXEC inquire/set for modenames	LOAD	ELPA	3	4048	-
DFHEIQSO		LOAD	ELPA	3		-
DFHEIQSP	EXEC inquire/set/discard for programs	LOAD	ELPA	3	6272	-
DFHEIQSQ	EXEC inquire/set for TD queues	LOAD	ELPA	3	9096	-
DFHEIQST	EXEC inquire/set for terminals	LOAD	ELPA	3	22480	-
DFHEIQSV	EXEC inquire/set for volumes	LOAD	ELPA	3	400	-
DFHEIQSX	EXEC inquire/set/discard for transactions	LOAD	ELPA	3	7144	-
DFHEIQSY		LOAD	ELPA	3		-
DFHEIQSZ	EXEC CICS SPI commands for FEPI	LOAD	ELPA	3	3784	-
DFHEIQTR	EXEC inquire/set for trace	LOAD	ELPA	3	10392	-
DFHEIQUE	EXEC inquire for exit programs	LOAD	ELPA	3	6288	-
DFHEIQVT	EXEC inquire/set for VTAM and autoinstall	LOAD	ELPA	3	5968	-
DFHEIQWB		LOAD	ELPA	3		-
DFHEIQWR		LOAD	ELPA	3		-
DFHEISO		LOAD	ELPA	3		-
DFHEISP	EXEC interface sycpoint processor	LOAD	ELPA	2	1576	-
I DFHEITAB	Translator table (basic commands)	LOAD	ELPA	3	46272	(1 on page 372)
I DFHEITBS	Translator table (special commands)	LOAD	ELPA	3	47984	(1 on page 372)
I DFHEITHG	EXEC interface hired gun lookup table	LOAD	LPA	2	13800	-
DFHEITMT	Command language table for CEMT	LOAD	ELPA	3	34432	(1 on page 372)

Table 28. LPA-eligible modules (continued)

Name	Description	Library	LPA/ ELPA	Pri	Size	Option/ Note
DFHEITOT	Command language table for CEOT	LOAD	ELPA	3	1192	(1 on page 372)
DFHEITS	EXEC TS request handler	LOAD	ELPA	2	7096	-
DFHEITST	CEST language definition table	LOAD	ELPA	3	4984	(1 on page 372)
DFHEITSZ	EXEC CICS language definition table	LOAD	ELPA	3	8592	(1 on page 372)
DFHEIWB		LOAD	ELPA	2		-
DFHEJC	EXEC interface for journaling	LOAD	ELPA	2	984	-
DFHEJDML		LOAD	ELPA	3		-
DFHEJITL		LOAD	ELPA	2		-
DFHEKC	EXEC interface for task control	LOAD	ELPA	2	1448	-
DFHEMDML		LOAD	ELPA	2		-
DFHEMEX	EXEC interface for ME domain	LOAD	ELPA	3	2792	-
DFHEMS	EXEC interface for BMS	LOAD	ELPA	2	4264	BMS
DFHEMTA	Programmable interface to Master terminal program	LOAD	ELPA	3	3232	(1 on page 372,5 on page 373)
DFHEMTD	Master terminal (CEMT) service program	LOAD	ELPA	3	98120	(1 on page 372)
DFHEMTP	Master terminal (CEMT) program	LOAD	ELPA	3	3232	(1 on page 372)
DFHEOP	EXEC interface for write operator	LOAD	ELPA	3	2752	-
DFHEOTP	CEOT service program	LOAD	ELPA	3	3232	(1 on page 372)
DFHEPC	EXEC interface for program control	LOAD	ELPA	2	8280	-
DFHEPS	System spooling interface stub	LOAD	ELPA	2	2856	(15 on page 374)
DFHERM	Resource manager interface (RMI) module	LOAD	LPA	2	14056	-
DFHERMRS	External resource manager resync processor	LOAD	ELPA	3	4312	-
DFHERMSP	External resource manager syncpoint processor	LOAD	ELPA	3	4400	-
DFHESC	EXEC interface for storage control	LOAD	ELPA	2	1320	-
DFHESE	EXEC interface for query security	LOAD	ELPA	2	4664	-
DFHESN	EXEC interface for signon and sign-off	LOAD	ELPA	2	4992	-
DFHESTP	CEST service program	LOAD	ELPA	3	3232	(1 on page 372)
DFHESZ	EXEC CICS API commands for FEPI	LOAD	ELPA	3	1136	-
DFHETC	EXEC interface for terminal control	LOAD	ELPA	2	7376	-
DFHETD	EXEC interface for transient data	LOAD	ELPA	2	2728	-
DFHETL	LU6.2 EXEC interface stub	LOAD	ELPA	2	7944	-
DFHETR	EXEC interface for trace control	LOAD	ELPA	2	792	(6 on page 373)
DFHETRX	EXEC interface for enter tracenum, monitor	LOAD	ELPA	2	1232	USERTR
DFHFCAT	File control catalog manager	LOAD	ELPA	2	7328	-

Table 28. LPA-eligible modules (continued)

Name	Description	Library	LPA/ ELPA	Pri	Size	Option/ Note
DFHFCBD	File control BDAM request processor	LOAD	LPA	2	5192	(16 on page 374)
DFHFCCA	File Control RLS Control ACB Manager	LOAD	ELPA	2	14984	-
DFHFCDL		LOAD	ELPA	2		-
DFHFCDN	File control DSN block manager	LOAD	ELPA	3	9856	(16 on page 374)
DFHFCDO		LOAD	ELPA	2		-
DFHFCDR		LOAD	ELPA	2		-
DFHFCDUW		LOAD	ELPA	2		-
DFHFCD2	File control shared data tables record request handler	LOAD	ELPA	2	16120	(16 on page 374) (10 on page 374)
DFHFCFL	File Control FRAB/FLAB processor	LOAD	ELPA	2	6896	-
DFHFCFR	File Control request handler	LOAD	ELPA	2	10760	(16 on page 374)
DFHFCFS	File Control state program	LOAD	ELPA	2	54632	(16 on page 374)
DFHFCIN	File control initialization program	LOAD	ELPA	3	1448	(16 on page 374)
DFHFCMJ	File control logging and journaling	LOAD	ELPA	2	16968	-
DFHFMT	File control table manager	LOAD	ELPA	3	11864	(16 on page 374)
DFHFCNQ	File control lock name interpreter	LOAD	ELPA	2	3088	-
DFHFCQI	File control VSAM RLS Quiesce - Initiation	LOAD	ELPA	2	5408	-
DFHFCQT	File control VSAM RLS Quiesce - Common System Transaction	LOAD	ELPA	2	14376	-
DFHFCQU	File control VSAM RLS Quiesce - Process	LOAD	ELPA	2	8304	-
DFHFCQX	File control VSAM RLS Quiesce - CICS RLS Quiesce Exit	LOAD	ELPA	2	2440	-
DFHFCRC	File control recovery control	LOAD	ELPA	2	20792	-
DFHFCRF		LOAD	ELPA	2		-
DFHFCRL	File control VSAM SHRCTL block manager	LOAD	ELPA	3	3320	(16 on page 374)
DFHFCRO		LOAD	ELPA	2	31384	-
DFHFCRP	File control restart program	LOAD	ELPA	3	24040	(16 on page 374)
DFHFCRS	File control VSAM RLS record management module	LOAD	ELPA	2	24192	-
DFHFCRV	File control VSAM RLS interface module	LOAD	ELPA	2	3576	-
DFHFCSD	File control shutdown program	LOAD	ELPA	2	1792	(16 on page 374)
DFHFCST	File control statistics program	LOAD	ELPA	3	7352	(16 on page 374)
DFHFCU	File open utility program	LOAD	LPA	3	552	(16 on page 374) (1 on page 372)

Table 28. LPA-eligible modules (continued)

Name	Description	Library	LPA/ ELPA	Pri	Size	Option/ Note
DFHFCVC		LOAD	ELPA	3		-
DFHFCVS	File access VSAM request processor	LOAD	ELPA	2	30504	(16 on page 374)
DFHGMM	VTAM LU startup message	LOAD	ELPA	2	2096	(1 on page 372)
DFHGTCNV	Subroutine used by the LOGR subsystem interface	LINK	ELPA	3	11736	-
DFHICP	Interval control program	LOAD	ELPA	2	11440	-
DFHICRC	Interval Control Recovery module	LOAD	ELPA	2	7616	-
DFHICXM	AP domain - bind, inquire, and release facility IC functions	LOAD	ELPA	2	4952	-
DFHIEDML		LOAD	ELPA	3		-
DFHIEP		LOAD	ELPA	3		-
DFHIEXM		LOAD	ELPA	2		-
DFHIIDML		LOAD	ELPA	3		-
DFHIIPA\$	BMS non-3270 input mapping (standard)	LOAD	ELPA	3	2048	(11 on page 374)
DFHIIP1\$	BMS non-3270 input mapping (full)	LOAD	ELPA	3	2048	(12 on page 374)
DFHIIXM		LOAD	ELPA	2		-
DFHINDAP	Indoubt tool	LOAD	ELPA	3	1912	(1 on page 372)
DFHINDSP	Indoubt tool syncpoint processor	LOAD	ELPA	3	1688	-
DFHINDT	Indoubt tool	LOAD	ELPA	3	6464	(1 on page 372)
DFHINTRU	Indoubt tool task-related user exit	LOAD	ELPA	3	2224	(1 on page 372)
DFHIRW10	IRC work delivery exit program	AUTH	ELPA	2	1328	-
DFHISCOP		LOAD	ELPA	3		-
DFHISDML		LOAD	ELPA	3		-
DFHISP	Intersystem communication program	LOAD	ELPA	2	3512	ISC=YES
DFHISRRP		LOAD	ELPA	3		-
DFHISXM		LOAD	ELPA	2		-
DFHJCICS		LOAD	ELPA	2		-
DFHJCP	Journaling component	LOAD	LPA	2	2288	-
DFHJVMAT		AUTH	ELPA	2		-
DFHKCP	Transaction manager startup routine	LOAD	ELPA	2	8936	-
DFHKCSC	DFHKCQ chain scanning for discard	LOAD	ELPA	3	1072	-
DFHKESVC	KE domain - authorized service routine	AUTH	ELPA	2	1768	-
DFHLDDMI	LD domain - secondary initialization	LOAD	ELPA	3	18480	-
DFHLDNT	LD domain - storage notify handler	LOAD	ELPA	2	2464	-
DFHLDST	LD domain - statistics collection	LOAD	ELPA	3	3592	-
DFHLDSVC	LD domain - authorized service routine	AUTH	LPA	2	2256	-
DFHLEDT	Transaction dump service	LOAD	ELPA	2	330	-
DFHLGCNV	Exit routine used by the LOGR subsystem interface	LINK	ELPA	3	20088	-

Table 28. LPA-eligible modules (continued)

Name	Description	Library	LPA/ ELPA	Pri	Size	Option/ Note
DFHLGDM	Log manager domain initialization	LOAD	ELPA	3	346736	-
DFHLIRET	Language interface return program	LOAD	LPA	2	136	-
DFHMCPA\$	BMS mapping control program (standard)	LOAD	ELPA	2	8248	(11 on page 374)
DFHMCPE\$	BMS mapping control program (minimum)	LOAD	ELPA	2	7064	(13 on page 374)
DFHMCP1\$	BMS mapping control program (full)	LOAD	ELPA	2	12920	(12 on page 374)
DFHMCX	BMS fast path module	LOAD	ELPA	2	7136	BMS
DFHMCY	Process MAPPINGEV Requests	LOAD	ELPA	2	7192	BMS
DFHMET1C		AUTH	ELPA	2		-
DFHMET1E	DFHMEU base messages link-edit module	AUTH	ELPA	2	340872	NATLANG
DFHMET1K		AUTH	ELPA	2		-
DFHMET5C		AUTH	ELPA	2		-
DFHMET5E	DFHMEU ONC RPS messages link-edit module	AUTH	ELPA	2	67512	-
DFHMET5K		AUTH	ELPA	2		-
DFHMET9C		AUTH	ELPA	2		-
DFHMET9E	DFHMEU user messages link-edit module	AUTH	ELPA	2	360	-
DFHMET9K		AUTH	ELPA	2		-
DFHMGP	Message writer program	LOAD	ELPA	3	14560	-
DFHMGT	Message generation table	LOAD	ELPA	3	23272	-
DFHMIRS	DFHMIRS	LOAD	ELPA	2	4816	ISC=YES (1 on page 372)
DFHML1	BMS LU1 printer mapping program	LOAD	ELPA	2	5144	BMS
DFHMNDML	MN domain - initialization/ termination	LOAD	ELPA	2	74936	-
DFHMNSVC	MN domain - authorized service routine	AUTH	ELPA	2	4600	-
DFHMQBAS	BMS program that handles the base panel	LOAD	ELPA	3	8432	(1 on page 372)
DFHMQBP0	CICS bridge - request handler	LOAD	ELPA	3	59608	(1 on page 372)
DFHMQBP1	CICS MQ bridge - DPL abend handler	LOAD	ELPA	3	31640	(1 on page 372)
DFHMQBR0	CICS MQ bridge - monitor main program	LOAD	ELPA	3	53816	(1 on page 372)
DFHMQCOD	PLT program to start connection using INITPARM	LOAD	ELPA	3	3968	(1 on page 372)
DFHMQCON	Back-end module that issues a connection request to the Websphere MQ subsystem	LOAD	ELPA	3	18576	(1 on page 372)
DFHMQCTL	First level transaction for operations. This program handles text mode operation only	LOAD	ELPA	3	6032	(1 on page 372)
DFHMQDCI	Data conversion exit for inbound data to the MQ-CICS bridge, for use in the CICS environment	LOAD	ELPA	3	4480	(1 on page 372)

Table 28. LPA-eligible modules (continued)

Name	Description	Library	LPA/ ELPA	Pri	Size	Option/ Note
DFHMQDCO	Data conversion exit for outbound data from the MQ-CICS bridge	LOAD	ELPA	3	3112	(1 on page 372)
DFHMQDIS	Back-end module that services display requests	LOAD	ELPA	3	15448	(1 on page 372)
DFHMQDSC	Back-end module that issues a disconnect request to the Websphere MQ subsystem	LOAD	ELPA	3	15600	(1 on page 372)
DFHMQDSL	Second level transaction for DISPLAY	LOAD	ELPA	3	6592	(1 on page 372)
DFHMQMON	Monitor program for pending events	LOAD	ELPA	3	6384	(1 on page 372)
DFHMQPLT	Sample CONNECT program in phase 2 PLT execution	LOAD	ELPA	3	640	(1 on page 372)
DFHMQPOP	BMS program that handles all pull-down maps	LOAD	ELPA	3	10936	(1 on page 372)
DFHMQPRM	Retrieves all default adapter parameters	LOAD	ELPA	3	1928	(1 on page 372)
DFHMQPUL	BMS program which handles all pull-down maps	LOAD	ELPA	3	11104	(1 on page 372)
DFHMQQCN	Second level transaction for CONNECT	LOAD	ELPA	3	8416	(1 on page 372)
DFHMQRET	BMS program that handles the final screen, including scrolling	LOAD	ELPA	3	10936	(1 on page 372)
DFHMQRS	Back-end module to service reset (change) requests	LOAD	ELPA	3	7528	(1 on page 372)
DFHMQSSQ	Back-end module to service start and stop CKTI requests	LOAD	ELPA	3	9256	(1 on page 372)
DFHMQTRU	CICS task-related user exit program for Websphere MQ	LOAD	ELPA	3	32384	(1 on page 372)
DFHMQTSK	Task initiation support (CKTI transaction)	LOAD	ELPA	3	14760	(1 on page 372)
DFHMRXM		LOAD	ELPA	2		-
DFHMSP	Message switching program	LOAD	ELPA	2	11440	(1 on page 372)
DFHMXP	Local queuing shipper	LOAD	ELPA	2	1176	(1 on page 372)
DFHM32A\$	BMS 3270 mapping (standard)	LOAD	ELPA	2	6520	(11 on page 374)
DFHM321\$	BMS 3270 mapping (full)	LOAD	ELPA	2	6520	(12 on page 374)
DFHNCIF		LINK	ELPA	3		-
DFHNQDM	NQ domain initialize, quiesce and terminate domain functions	LOAD	ELPA	2	34760	-
DFHOTDML		LOAD	ELPA	2		-
DFHPBPA\$	BMS page and text build (standard)	LOAD	ELPA	2	8552	(11 on page 374)
DFHPBP1\$	BMS page and text build (full)	LOAD	ELPA	2	9352	(12 on page 374)
DFHPCP	Program control program	LOAD	ELPA	2	2488	-
DFHPGADX	Program autoinstall exit - Assembler	LOAD	ELPA	2	200	(1 on page 372)



Table 28. LPA-eligible modules (continued)

Name	Description	Library	LPA/ELPA		Pri	Size	Option/Note
DFHPGDM	PG domain - initialize, quiesce, and terminate domain functions	LOAD	ELPA		2	153328	-
DFHPGRP	PG domain - recovery program	LOAD	ELPA		2	12888	-
DFHPHP	Partition handling program	LOAD	ELPA		2	2248	BMS
DFHPIAP		LOAD	ELPA		2		-
DFHPIDML		LOAD	ELPA		2		-
DFHPIDSH		LOAD	ELPA		3		-
DFHPIDSQ		LOAD	ELPA		3		-
DFHPIEP		LOAD	ELPA		2		-
DFHPIITL		LOAD	ELPA		2		-
DFHPILSQ		LOAD	ELPA		3		-
DFHPIPA		LOAD	ELPA		2		-
DFHPIRI		LOAD	ELPA		3		-
DFHPIRS		LOAD	ELPA		3		-
DFHPIRT		LOAD	ELPA		3		-
DFHPISN1		LOAD	ELPA		2		-
DFHPISN2		LOAD	ELPA		2		-
DFHPITP		LOAD	ELPA		3		-
DFHPITQ1		LOAD	ELPA		3		-
DFHPIXC		LOAD	ELPA		2		-
DHFPIXE		LOAD	ELPA		2		-
DFHPRCM	Partner resource manager command interface	LOAD	ELPA		3	1360	-
DFHPRFS	Partner resource manager interface to SAA communications interface	LOAD	ELPA		2	632	-
DFHPRIN	Partner initialization load program	LOAD	ELPA		3	3336	-
DFHPRPT	Partner resource table (PRT) manager	LOAD	ELPA		3	3040	-
DFHPSP	System spooling interface program	LOAD	LPA		2	14424	(15 on page 374)
DFHPSSVC	System spooling interface, retrieve a data set name	AUTH	ELPA		2	1472	(15 on page 374)
DFHPTDML		LOAD	ELPA		3		-
DFHQRY	Query transaction	LOAD	ELPA		2	3824	(1 on page 372)
DFHRLRA\$	BMS route list resolution (standard)	LOAD	ELPA		2	2040	(11 on page 374)
DFHRLR1\$	BMS route list resolution (full)	LOAD	ELPA		2	3832	(12 on page 374)
DFHRMSY	Resource manager resync program	LOAD	ELPA		3	4160	(1 on page 372)
DFHRPAL	ONC RPC feature alias list	LOAD	ELPA		3	2384	(1 on page 372)
DFHRTC	CRTE cancel command processor	LOAD	ELPA		2	864	(1 on page 372)
DFHRTE	Transaction routing program	LOAD	ELPA		2	2600	(1 on page 372)
DFHRTSU	Surrogate terminal interface program	LOAD	ELPA		3	2736	-
DFHRXDML		LOAD	ELPA		2		-
DFHRZDML		LOAD	ELPA		2		-

Table 28. LPA-eligible modules (continued)

Name	Description	Library	LPA/ ELPA	Pri	Size	Option/ Note
DFHRZJN		LOAD	ELPA	2		-
DFHSAIQ	AP domain - system data inquire & set	LOAD	ELPA	2	2224	-
DFHSFP	Sign-off program	LOAD	ELPA	2	4144	(1 on page 372)
DFHSHDML		LOAD	ELPA	2		-
DFHSHRRP		LOAD	ELPA	2		-
DFHSHRSP		LOAD	ELPA	2		-
DFHSHSY		LOAD	ELPA	2		-
DFHSJCSL		AUTH	ELPA	3		-
DFHSJDML		LOAD	ELPA	3		-
DFHSJLL		AUTH	ELPA	3		-
DFHSJML		LOAD	ELPA	3		-
DFHSKP	Subtask management program	LOAD	ELPA	2	6448	-
DFHSKTSK	General purpose subtask entry point	AUTH	ELPA	3	40	-
DFHSMSVC	SM domain - authorized service routine	AUTH	ELPA	3	11336	-
DFHSMTAB	CICSplex SM commands language table	LOAD	ELPA	3	632	-
DFHSNP	Signon program	LOAD	ELPA	2	13184	(1 on page 372)
DFHSNUS	US domain - local and remote signon	LOAD	ELPA	2	52984	-
DFHSODML		LOAD	ELPA	2		-
DFHSEOSE		LOAD	ELPA	3		-
DFHSPP	Syncpoint program	LOAD	ELPA	2	1744	-
DFHSTDML	ST domain - initialization/termination	LOAD	ELPA	3	30520	-
DFHSUSX	XRF signon	LOAD	ELPA	2	9240	(19 on page 374)
DFHSUWT	WTO/WTOR interface subroutine	LOAD	ELPA	3	7152	-
DFHSUZX	ZC trace controller	LOAD	ELPA	3	6808	-
DFHSZATR	FEPI adaptor program	LOAD	ELPA	3	17328	-
DFHTBSS	Builder syncpoint processor	LOAD	ELPA	2	19168	-
DFHTCRP	Terminal control recovery program	LOAD	ELPA	3	25392	-
DFHTDP	Transient data program	LOAD	LPA	2	25448	-
DFHTDQ	Transient data program	LOAD	ELPA	2	30280	-
DFHTDRM	Transient data recovery manager processor	LOAD	ELPA	2	31080	-
DFHTDRP	Transient data recovery program	LOAD	ELPA	3	6432	-
DFHTDTM	TD table management gate	LOAD	ELPA	2	17656	-
DFHTDXM	XM domain - TD facility management services	LOAD	ELPA	2	3696	-
DFHTFBF	Terminal facility manager bind facility functions	LOAD	ELPA	2	14528	-
DFHTFIQ	Terminal facility manager inquire/set functions	LOAD	ELPA	2	5632	-
DFHTFRF	Terminal facility manager release function	LOAD	ELPA	2	4176	-
DFHTFXM		LOAD	ELPA	2		-

Table 28. LPA-eligible modules (continued)

Name	Description	Library			Size	Option/ Note
			LPA/ ELPA	Pri		
DFHTIDM	TI domain - initialization/ termination	LOAD	ELPA	3	9264	-
DFHTMP	Table manager program	LOAD	ELPA	2	21088	-
DFHTON	Terminal object resolution module	LOAD	ELPA	2	816	-
DFHTONR	Terminal object resolution recovery	LOAD	ELPA	2	2248	-
DFHTORP	Terminal object recovery program	LOAD	LPA	3	544	-
DFHTPPA\$	BMS terminal page processor (standard)	LOAD	ELPA	2	3344	(11 on page 374)
DFHTPP1\$	BMS terminal page processor (full)	LOAD	ELPA	2	4336	(12 on page 374)
DFHTPQ	BMS terminal page cleanup program	LOAD	ELPA	2	4040	BMS (1 on page 372)
DFHTPR	BMS terminal page retrieval program	LOAD	ELPA	2	21688	BMS (1 on page 372)
DFHTPS	BMS terminal page scheduling program	LOAD	ELPA	2	4632	BMS (1 on page 372)
DFHTRAO	TR domain - auxiliary trace output	LOAD	LPA	3	1480	(20 on page 374)
DFHTSDML	Temporary storage domain	LOAD	ELPA	2	123248	-
DFHTSDQ		LOAD	ELPA	2		-
DFHTSP	Temporary-storage control program	LOAD	ELPA	2	3928	(21 on page 374)
DFHUEH	User exit handler (AP domain)	LOAD	ELPA	2	7960	-
DFHUEM	User exit manager	LOAD	ELPA	3	7864	-
DFHUSDM	US domain - initialize, quiesce, and terminate domain functions	LOAD	ELPA	3	58640	-
DFHWBAAX		LOAD	ELPA	2		-
DFHWBADX		LOAD	ELPA	2		-
DFHWBDML		LOAD	ELPA	2		-
DFHWBENV		LOAD	ELPA	3		-
DFHWBERX		LOAD	ELPA	2		-
DFHWBIMG		LOAD	ELPA	3		-
DFHWBPA		LOAD	ELPA	3		-
DFHWBUN		LOAD	ELPA	3		-
DFHWSMS	DFHWSMS	AUTH	ELPA	2	38456	(19 on page 374)
DFHWSSON	CAVM state management signon request handler	AUTH	ELPA	2	14280	(19 on page 374)
DFHWTI	XRF takeover initiation program	AUTH	ELPA	3	11648	(19 on page 374)
DFHXCEIX		EXCI	ELPA	2		-
DFHXCI	External CICS interface (EXCI) program	LOAD	LPA	3	3152	-
DFHXCPRX		EXCI	ELPA	2		-
DFHXCSVC	EXCI SVC services	LINK	ELPA	3	512	-
DFHXCTAB	EXCI language table	LOAD	ELPA	3	504	-
DFHXFP	Online data transformation program	LOAD	ELPA	2	31744	ISC=YES
DFHXFRM	Function shipping storage recovery	LOAD	ELPA	2	1744	-
DFHXFX	Optimized data transformation program	LOAD	ELPA	2	8024	ISC=YES

Table 28. LPA-eligible modules (continued)

Name	Description	Library	LPA/ ELPA	Pri	Size	Option/ Note
DFHXRP	XRF request program	LOAD	ELPA	2	9272	(19 on page 374)
DFHXRSP	XRF surveillance program	LOAD	ELPA	2	4800	(19 on page 374)
DFHXSS	XS domain - supervisor request services	AUTH	ELPA	3	30576	SEC=NO
DFHXSWM	XRF message manager for security manager	LOAD	ELPA	2	1744	(19 on page 374)
DFHXTP	Terminal sharing transformation program	LOAD	ELPA	2	11656	ISC=YES
DFHZATA	Autoinstall program	LOAD	ELPA	2	18648	(1 on page 372)
DFHZATD	Autoinstall delete program	LOAD	ELPA	2	6584	(1 on page 372)
DFHZATDX	User-replaceable autoinstall exit	LOAD	ELPA	2	392	AIEXIT (1 on page 372)
DFHZATDY	User-replaceable autoinstall exit with APPC	LOAD	ELPA	2	560	AIEXIT (1 on page 372)
I DFHZBAN	Terminal control bind analysis	LOAD	LPA	2	10288	-
DFHZCA	VTAM working set module	LOAD	ELPA	2	9888	(18 on page 374)
DFHZCB	VTAM working set module	LOAD	ELPA	2	39496	(18 on page 374)
DFHZCC	VTAM working set module	LOAD	ELPA	2	63160	(18 on page 374)
DFHZCN1	CICS Client CCIN transaction	LOAD	ELPA	3	4472	(1 on page 372)
DFHZCN2	CICS Client CCIN transaction	LOAD	ELPA	3	4464	-
DFHZCP	Terminal management program	LOAD	ELPA	2	33528	(18 on page 374)
DFHZCT1	CICS Client CTIN transaction	LOAD	ELPA	3	103046	(1 on page 372)
DFHZCUT	Persistent verification signed-on-from list management program	LOAD	ELPA	2	5376	(18 on page 374)
DFHZCW	VTAM nonworking set module	LOAD	ELPA	3	7072	(18 on page 374)
DFHZCX	LOCATE, ISC/IRC request	LOAD	ELPA	2	34728	ISC=YES
DFHZCXR	Transaction routing module address list	LOAD	ELPA	2	28984	ISC=YES
DFHZCY	VTAM nonworking set module	LOAD	ELPA	3	82976	(18 on page 374)
DFHZCZ	VTAM nonworking set module	LOAD	ELPA	3	25712	(18 on page 374)
DFHZGAI	APPC autoinstall - create APPC clones	LOAD	ELPA	2	9288	AIEXIT
DFHZGBM	APPC manipulate bitmap	LOAD	ELPA	2	4776	(18 on page 374)
DFHZGCA	LU6.2 CNOS actioning	LOAD	ELPA	3	6168	(18 on page 374)

Table 28. LPA-eligible modules (continued)

Name	Description	Library	LPA/ ELPA	Pri	Size	Option/ Note
DFHZGCC	Catalog CNOS services	LOAD	ELPA	3	2440	(18 on page 374)
DFHZGCH	ZC VTAM change macro domain function	LOAD	ELPA	3	4056	-
DFHZGCN	LU6.2 CNOS negotiation	LOAD	ELPA	3	12272	(18 on page 374)
DFHZGIN	ZC VTAM issue inquire	LOAD	ELPA	3	3544	-
DFHZGPR	VTAM persistent sessions resource handler	LOAD	ELPA	3	2848	(18 on page 374)
DFHZGTA	ZC table alter	LOAD	ELPA	2	23312	-
DFHZGTI	ZC table inquire gate	LOAD	ELPA	2	14744	-
DFHZHPRX	Authorized path SRB mode VTAM EXECRPL	AUTH	ELPA	2	712	HPO=YES
DFHZLS1	LU6.2 CNOS request transaction program	LOAD	ELPA	3	2160	(18 on page 374) (1 on page 372)
DFHZRSP	Resync send program	LOAD	ELPA	2	248	(22 on page 374) (1 on page 372)
DFH62XM		LOAD	ELPA	2		-
ICCFCTAB		LOAD	ELPA	3		-
ICCFCTBH		LOAD	ELPA	3		-



---

## Appendix B. CICSplex SM modules eligible for the MVS link pack area

CICSplex SM modules that are eligible for installation in the MVS link pack area (LPA) are specified in the CICSplex SM supplied USERMOD: EYU\$UM01.

This following table lists the CICSplex SM modules that are eligible for inclusion in the LPA.

You must allow enough space in the link pack area for the installation of the selected CICSplex SM modules.

The total space needed depends on how the modules are packaged into the link pack area by the operating system and a local MAS requires approximately 2034KB.

All CICSplex SM modules are installed in the LOAD library and loaded in the ELPA.

*Table 29. CICSplex SM modules eligible for the LPA*

<b>Module</b>	<b>Description</b>
EYU9BA01	BAS MAS object
EYU9CM01	MAS communications
EYU9MN01	MAS monitor
EYU9NA01	MAS agents
EYU9PS01	MAS real-time analysis
EYU9TS01	MAS topology
EYU9WM01	MAS workload management
EYU9XC01	MAS cache
EYU9XD01	MAS data repository
EYU9XL01	MAS kernel linkage
EYU9XM01	MAS message format
EYU9XQ01	MAS queue manager
EYU9XS01	MAS common services
EYU9XZ01	MAS trace





---

## Appendix C. Default CICS resource definitions for CICSplex SM

These tables describe the default CICS resource definitions supplied for CICSplex SM to start a CMAS, MAS, and WUI.

### Default CICS resource definitions for a CMAS

The default CICS resource definitions supplied for CICSplex SM to start a CMAS, MAS, and WUI. These are supplied in samples EYU\$CDEF, EYU\$MDEF, and EYU\$WDEF in the SEYUSAMP library. This topic lists them.

On an INITIAL start of a CMAS, CICSplex SM uses the default CICS resource definitions supplied in sample EYU\$CDEF. Table 30 lists these definitions.

*Table 30. Sample resource definitions supplied in EYU\$CDEF*

Resource type	Name	Description
File	EYUDWREP	CICSplex SM data repository
	EYUDREP	CICSplex SM data repository, alternate definition
Profile	EYUCICSD	CICSplex SM debugger

Table 30. Sample resource definitions supplied in EYU\$CDEF (continued)

Resource type	Name	Description
Program	EYUTMMDT	CICSplex SM monitor definition table
	EYUTXDDA	CICSplex SM repository record control table
	EYUTXLNT	CICSplex SM notification table
	EYUTXLPD	CICSplex SM parameter table
	EYU9BA00	CICSplex SM BAS object
	EYU9CM00	CICSplex SM CMAS communications
	EYU9DBG0	CICSplex SM debugger
	EYU9DBG1	CICSplex SM debugger
	EYU9DBG2	CICSplex SM debugger
	EYU9DEBUG	CICSplex SM debugger
	EYU9MN00	CICSplex SM MAS monitor
	EYU9NA00	CICSplex SM CMAS MAS component
	EYU9NA01	CICSplex SM batch relation table
	EYU9NLDO	CICSplex SM MAS XDUOUT exit
	EYU9NXOP	CICSplex SM CMAS MAS program
	EYU9PS00	CICSplex SM CMAS real-time analysis (RTA)
	EYU9TS00	CICSplex SM CMAS topology
	EYU9WM00	CICSplex SM CMAS workload management
	EYU9XC00	CICSplex SM CMAS cache
	EYU9XDBU	CICSplex SM CMAS batch repository input/output utility
	EYU9XD00	CICSplex SM CMAS data repository
	EYU9XLCI	CICSplex SM EUI selector to CMAS client
	EYU9XLCS	CICSplex SM PLTPI startup program
	EYU9XLEV	CICSplex SM main initialization
	EYU9XLGR	CICSplex SM global user abend exit
	EYU9XLME	CICSplex SM message exit
	EYU9XLOP	CICSplex SM object environment CREATE
	EYU9XLSR	CICSplex SM system user abend exit
	EYU9XL00	CICSplex SM CMAS kernel linkage
	EYU9XM00	CICSplex SM CMAS message format
EYU9XQ00	CICSplex SM queue manager	
EYU9XSTR	CICSplex SM CMAS-MAS task recovery task related user exit	
EYU9XS00	CICSplex SM common services	
EYU9XZ00	CICSplex SM CMAS trace	

Table 30. Sample resource definitions supplied in EYU\$CDEF (continued)

Resource type	Name	Description
Transaction	BMLT	CICSplex SM BAS long-running task
	CODB	CICSplex SM debugger
	COD0	CICSplex SM debugger
	COD1	CICSplex SM debugger
	COD2	CICSplex SM debugger
	COLU	CICSplex SM CICS online utility
	COSD	CICSplex SM CMAS stop
	LCMU	CICSplex SM Transmit must-complete remote message argument lists
	LCPP	CICSplex SM check joining CMAS for plex export
	LECI	CICSplex SM CMAS-MAS CPI-C initial contact - CMAS
	LECR	CICSplex SM CMAS-MAS CPI-C RECEIVE manager - CMAS
	LECS	CICSplex SM CMAS-MAS CPI-C SEND manager - CMAS
	LEEI	CICSplex SM ESSS initial contact transient
	LEER	CICSplex SM ESSS RECEIVE link manager
	LEMI	CICSplex SM CMAS-MAS MRO initial contact - CMAS
	LEMS	CICSplex SM CMAS-MAS MRO SEND manager - CMAS
	LENS	CICSplex SM CMAS-NetView SEND link manager
	LMIR	CICSplex SM MRO RECEIVE link - CMAS
	LNCI	CICSplex SM CMAS-CMAS CPI-C initial contact
	LNCS	CICSplex SM CMAS-CMAS CPI-C SEND link manager
	LNMI	CICSplex SM CMAS-CMAS initial contact task
	LNMS	CICSplex SM CMAS-CMAS MRO initial contact task
	LPDG	CICSplex SM net direct address generation
	LPLK	CICSplex SM network lock
	LPLT	CICSplex SM connection long-running task??
	LPRT	CICSplex SM connection services remove link??
	LPSC	CICSplex SM repository synchronization check
	LPSM	CICSplex SM CMAS - execute MAS shutdown
	LRLT	CICSplex SM security long-running task
	LSGT	CICSplex SM garbage collection long-running task
	LSRT	CICSplex SM communications long-running task
	LWTM	CICSplex SM timing transaction
	MCCM	CICSplex SM data collector manager
	MCTK	CICSplex SM monitor end-of-task data collector
	MMEI	CICSplex SM end-of-interval processor
	MMIS	CICSplex SM monitor MAS attach
	MMST	CICSplex SM monitor initialization
	PEAD	CICSplex SM RTA evaluation asynchronous data processor
	PELT	CICSplex SM RTA evaluation long-running task

Table 30. Sample resource definitions supplied in EYU\$CDEF (continued)

Resource type	Name	Description
Transaction	PMLT	CICSplex SM RTA MRM long-running task
	PNLT	CICSplex SM RTA ACT long-running task
	PPLT	CICSplex SM RTA analysis point long-running task
	PRLT	CICSplex SM RTA ANL long-running task
	PRPR	CICSplex SM RTA analysis asynchronous data processor
	PSLT	CICSplex SM POL long-running transaction
	TICT	CICSplex SM topology long-running task map change
	TIRT	CICSplex SM topology long-running task RODM
	TIST	CICSplex SM topology long-running task services
	TSMH	CICSplex SM ??
	TSPD	CICSplex SM topology CICSplex delete
	TSSC	CICSplex SM topology system start event
	TSSJ	CICSplex SM topology join event
	WMCC	CICSplex SM workload manager (WLM) CRE ?? work
	WMGR	CICSplex SM WLM LTR ?? transaction
	WMLA	CICSplex SM WLM update AOR abend lists
	WMQB	CICSplex SM query workloads for a CICSplex
	WMQM	CICSplex SM manage query workload process
	WMQS	CICSplex SM quiescing AOR task
	WMSC	CICSplex SM update scope service class tables
	WMWC	CICSplex SM WLM CRE ?? work
	WMWT	CICSplex SM WLM TRM ?? work
	WSCL	CICSplex SM global lock service
	WSLW	CICSplex SM locking task
	XDBM	CICSplex SM batch repository mainline
	XDNC	CICSplex SM API notification long-running task
	XDND	CICSplex SM API disposition long-running task
	XDNE	CICSplex SM API connection long-running task
	XDNR	CICSplex SM API command-processing task
	XDNS	CICSplex SM API service long-running task
	XDSR	CICSplex SM build MPSYNCCR ?? record queue
	XLEV	CICSplex SM initialization transaction
	XLNX	CICSplex SM notify long-running task
XLST	CICSplex SM selector transaction	
XMLT	CICSplex SM consolidated message log long-running task	
XQST	CICSplex SM asynchronous spooling of CICSplex SM queue	

Table 30. Sample resource definitions supplied in EYU\$CDEF (continued)

Resource type	Name	Description
Transient data queue	COLG	CICSplex SM log output
	COPR	CICSplex SM parameters

### Default CICS resource definitions for a MAS

On an INITIAL start of a MAS, or if a MAS is started by the COLM transaction, CICSplex SM uses the default CICS resource definitions supplied in sample EYU\$MDEF. Table 31 lists these definitions.

Table 31. Sample resource definitions supplied in EYU\$MDEF

Resource type	Name	Description
Profile	EYUCICSD	CICSplex SM debugger transaction

Table 31. Sample resource definitions supplied in EYU\$MDEF (continued)

Resource type	Name	Description
Program	EYUTXLPD	CICSPlex SM parameter table
	EYU9AEDF	CICSPlex SM API EDF formatter
	EYU9BA01	CICSPlex SM MAS BAS object
	EYU9CM01	CICSPlex SM CMAS communications
	EYU9DBG0	CICSPlex SM debugger
	EYU9DBG1	CICSPlex SM debugger
	EYU9DBG2	CICSPlex SM debugger
	EYU9DEBUG	CICSPlex SM debugger
	EYU9MN01	CICSPlex SM MAS monitor
	EYU9NA01	CICSPlex SM MAS agents
	EYU9NLDC	CICSPlex SM MAS XDUREQC exit
	EYU9NLDO	CICSPlex SM MAS XDUOUT exit
	EYU9NLDR	CICSPlex SM MAS XDUREQ exit
	EYU9NLID	CICSPlex SM MAS XRSINDI exit
	EYU9NLME	CICSPlex SM MAS XMEOUT exit
	EYU9NLSO	CICSPlex SM MAS XSNOFF exit
	EYU9NMST	CICSPlex SM MAS XSTOUT exit
	EYU9NMTE	CICSPlex SM MAS XMNOUT exit
	EYU9NPS2	CICSPlex SM MAS shutdown task
	EYU9NXLM	CICSPlex SM MAS PLTPI
	EYU9NXSD	CICSPlex SM MAS global shutdown exit
	EYU9NXSH	CICSPlex SM MAS global shutdown program
	EYU9PS01	CICSPlex SM MAS real-time analysis
	EYU9TS01	CICSPlex SM MAS topology
	EYU9WM01	CICSPlex SM MAS workload management
	EYU9WRAM	CICSPlex SM WLM route
	EYU9XC01	CICSPlex SM MAS cache
	EYU9XC02	CICSPlex SM MAS cache
	EYU9XD01	CICSPlex SM MAS data repository
	EYU9XLAP	CICSPlex SM API object environment CREATE
	EYU9XLEV	CICSPlex SM main initialization
	EYU9XLGR	CICSPlex SM global user abend XPCTA exit
	EYU9XLOP	CICSPlex SM object environment CREATE
	EYU9XLSR	CICSPlex SM system user abend XSRAB exit
	EYU9XL01	CICSPlex SM MAS kernel linkage
	EYU9XM01	CICSPlex SM MAS message format
	EYU9XQ01	CICSPlex SM MAS queue manager
	EYU9XSTR	CICSPlex SM CMAS-MAS task recovery task related user exit
	EYU9XS01	CICSPlex SM MAS common services
	EYU9XZ01	CICSPlex SM MAS trace

Table 31. Sample resource definitions supplied in EYU\$MDEF (continued)

Resource type	Name	Description
Transaction	CODB	CICSplex SM debugger
	COD0	CICSplex SM debugger
	COD1	CICSplex SM debugger
	COD2	CICSplex SM debugger
	COIE	CICSplex SM MAS status transaction
	COHT	CICSplex SM MAS historical data collection transaction
	COIR	CICSplex SM RTA evaluation definition task - MAS
	COI0	CICSplex SM MAS communications receive transaction
	COLM	CICSplex SM MAS startup
	COLU	CICSplex SM CICS online utility
	CONA	CICSplex SM MAS alternate long-running task transaction
	COND	CICSplex SM MAS shutdown transaction
	CONH	CICSplex SM MAS history task
	CONL	CICSplex SM MAS initialization transaction
	CONM	CICSplex SM MAS monitor task
	CORT	CICSplex SM RTA task
	COSH	CICSplex SM MAS shutdown
COWC	CICSplex SM MAS workload management garbage collection	
Transient data queue	COPR	CICSplex SM buffers

## Default CICS resource definitions for a WUI

On an INITIAL start of a WUI, or if a WUI is started by the COVC transaction, CICSplex SM uses the default CICS resource definitions supplied in sample EYU\$WDEF. Table 32 lists these definitions. As a WUI server is a MAS, CICSplex SM also uses the default CICS resource definitions supplied in sample EYU\$MDEF, and listed in Table 31 on page 399.

Table 32. Sample resource definitions supplied in EYU\$WDEF

Resource type	Name	Description
Enqmodel	EYUWREP	CICSplex SM WUI repository update serialization
File	EYUWREP	CICSplex SM WUI data repository
Mapset	EYU9VCE	CICSplex SM WUI control mapset

Table 32. Sample resource definitions supplied in EYU\$WDEF (continued)

Resource type	Name	Description
Program	EYU9VKEC	CICSPlex SM WUI server
	EYU9VKIT	CICSPlex SM WUI initialization
	EYU9VWAN	CICSPlex SM WUI analyzer
	EYU9VWCV	CICSPlex SM WUI converter
	EYUTVOSE	CICSPlex SM WUI resource table text strings (US English)
	EYUTVTGE	CICSPlex SM WUI GIF files (US English)
	EYUTVTHE	CICSPlex SM WUI HTML pages (US English)
	EYUTVTJE	CICSPlex SM Java classes (US English)
	EYUTVTME	CICSPlex SM WUI messages (US English)
	EYUTVTSE	CICSPlex SM WUI strings (US English)
	EYUTVOSK	CICSPlex SM WUI resource table text strings (Japanese)
	EYUTVTGK	CICSPlex SM WUI GIF files (Japanese)
	EYUTVTHK	CICSPlex SM WUI HTML pages (Japanese)
	EYUTVTJK	CICSPlex SM Java classes (Japanese)
	EYUTVTMK	CICSPlex SM WUI messages (Japanese)
	EYUTVTSK	CICSPlex SM WUI strings (Japanese)
	EYUTVOSS	CICSPlex SM WUI resource table text strings (Simplified Chinese)
	EYUTVTGS	CICSPlex SM WUI GIF files (Simplified Chinese)
	EYUTVTHS	CICSPlex SM WUI HTML pages (Simplified Chinese)
	EYUTVTJS	CICSPlex SM Java classes (Simplified Chinese)
EYUTVTMS	CICSPlex SM WUI messages (Simplified Chinese)	
EYUTVTSS	CICSPlex SM WUI strings (Simplified Chinese)	
Transaction	COVA	CICSPlex SM WUI user API task
	COVC	CICSPlex SM WUI server controller
	COVE	CICSPlex SM WUI HTTP error handler
	COVG	CICSPlex SM WUI global task
	COVP	CICSPlex SM WUI proxy
	COVU	CICSPlex SM WUI resource server
Transient data queue	COLG	CICSPlex SM log output
	COVE	CICSPlex SM WUI export data set
	COVI	CICSPlex SM WUI import data set
	COVP	CICSPlex SM WUI parameters



# Bibliography

## The CICS Transaction Server for z/OS library

The published information for CICS Transaction Server for z/OS is delivered in the following forms:

### The CICS Transaction Server for z/OS Information Center

The CICS Transaction Server for z/OS Information Center is the primary source of user information for CICS Transaction Server. The Information Center contains:

- Information for CICS Transaction Server in HTML format.
- Licensed and unlicensed CICS Transaction Server books provided as Adobe Portable Document Format (PDF) files. You can use these files to print hardcopy of the books. For more information, see “PDF-only books.”
- Information for related products in HTML format and PDF files.

One copy of the CICS Information Center, on a CD-ROM, is provided automatically with the product. Further copies can be ordered, at no additional charge, by specifying the Information Center feature number, 7014.

Licensed documentation is available only to licensees of the product. A version of the Information Center that contains only unlicensed information is available through the publications ordering system, order number SK3T-6945.

### Entitlement hardcopy books

The following essential publications, in hardcopy form, are provided automatically with the product. For more information, see “The entitlement set.”

## The entitlement set

The entitlement set comprises the following hardcopy books, which are provided automatically when you order CICS Transaction Server for z/OS, Version 3 Release 2:

*Memo to Licensees*, GI10-2559  
*CICS Transaction Server for z/OS Program Directory*, GI13-0515  
*CICS Transaction Server for z/OS Release Guide*, GC34-6811  
*CICS Transaction Server for z/OS Installation Guide*, GC34-6812  
*CICS Transaction Server for z/OS Licensed Program Specification*, GC34-6608

You can order further copies of the following books in the entitlement set, using the order number quoted above:

*CICS Transaction Server for z/OS Release Guide*  
*CICS Transaction Server for z/OS Installation Guide*  
*CICS Transaction Server for z/OS Licensed Program Specification*

## PDF-only books

The following books are available in the CICS Information Center as Adobe Portable Document Format (PDF) files:

### CICS books for CICS Transaction Server for z/OS

#### General

*CICS Transaction Server for z/OS Program Directory*, GI13-0515  
*CICS Transaction Server for z/OS Release Guide*, GC34-6811  
*CICS Transaction Server for z/OS Migration from CICS TS Version 3.1*, GC34-6858

*CICS Transaction Server for z/OS Migration from CICS TS Version 1.3,*  
GC34-6855

*CICS Transaction Server for z/OS Migration from CICS TS Version 2.2,*  
GC34-6856

*CICS Transaction Server for z/OS Installation Guide,* GC34-6812

#### **Administration**

*CICS System Definition Guide,* SC34-6813

*CICS Customization Guide,* SC34-6814

*CICS Resource Definition Guide,* SC34-6815

*CICS Operations and Utilities Guide,* SC34-6816

*CICS Supplied Transactions,* SC34-6817

#### **Programming**

*CICS Application Programming Guide,* SC34-6818

*CICS Application Programming Reference,* SC34-6819

*CICS System Programming Reference,* SC34-6820

*CICS Front End Programming Interface User's Guide,* SC34-6821

*CICS C++ OO Class Libraries,* SC34-6822

*CICS Distributed Transaction Programming Guide,* SC34-6823

*CICS Business Transaction Services,* SC34-6824

*Java Applications in CICS,* SC34-6825

*JCICS Class Reference,* SC34-6001

#### **Diagnosis**

*CICS Problem Determination Guide,* SC34-6826

*CICS Messages and Codes,* GC34-6827

*CICS Diagnosis Reference,* GC34-6862

*CICS Data Areas,* GC34-6863-00

*CICS Trace Entries,* SC34-6828

*CICS Supplementary Data Areas,* GC34-6864-00

#### **Communication**

*CICS Intercommunication Guide,* SC34-6829

*CICS External Interfaces Guide,* SC34-6830

*CICS Internet Guide,* SC34-6831

#### **Special topics**

*CICS Recovery and Restart Guide,* SC34-6832

*CICS Performance Guide,* SC34-6833

*CICS IMS Database Control Guide,* SC34-6834

*CICS RACF Security Guide,* SC34-6835

*CICS Shared Data Tables Guide,* SC34-6836

*CICS DB2 Guide,* SC34-6837

*CICS Debugging Tools Interfaces Reference,* GC34-6865

### **CICSplex SM books for CICS Transaction Server for z/OS**

#### **General**

*CICSplex SM Concepts and Planning,* SC34-6839

*CICSplex SM User Interface Guide,* SC34-6840

*CICSplex SM Web User Interface Guide,* SC34-6841

#### **Administration and Management**

*CICSplex SM Administration,* SC34-6842

*CICSplex SM Operations Views Reference,* SC34-6843

*CICSplex SM Monitor Views Reference,* SC34-6844

*CICSplex SM Managing Workloads,* SC34-6845

*CICSplex SM Managing Resource Usage,* SC34-6846

*CICSplex SM Managing Business Applications,* SC34-6847

#### **Programming**

*CICSplex SM Application Programming Guide,* SC34-6848

*CICSplex SM Application Programming Reference,* SC34-6849

## Diagnosis

*CICSplex SM Resource Tables Reference*, SC34-6850  
*CICSplex SM Messages and Codes*, GC34-6851  
*CICSplex SM Problem Determination*, GC34-6852

## CICS family books

### Communication

*CICS Family: Interproduct Communication*, SC34-6853  
*CICS Family: Communicating from CICS on zSeries*, SC34-6854

### Licensed publications

The following licensed publications are not included in the unlicensed version of the Information Center:

*CICS Diagnosis Reference*, GC34-6862  
*CICS Data Areas*, GC34-6863-00  
*CICS Supplementary Data Areas*, GC34-6864-00  
*CICS Debugging Tools Interfaces Reference*, GC34-6865

---

## Other CICS books

The following publications contain further information about CICS, but are not provided as part of CICS Transaction Server for z/OS, Version 3 Release 2.

<i>Designing and Programming CICS Applications</i>	SR23-9692
<i>CICS Application Migration Aid Guide</i>	SC33-0768
<i>CICS Family: API Structure</i>	SC33-1007
<i>CICS Family: Client/Server Programming</i>	SC33-1435
<i>CICS Transaction Gateway for z/OS Administration</i>	SC34-5528
<i>CICS Family: General Information</i>	GC33-0155
<i>CICS 4.1 Sample Applications Guide</i>	SC33-1173
<i>CICS/ESA 3.3 XRF Guide</i>	SC33-0661

---

## Books from related libraries

### Systems Network Architecture (SNA)

- *z/OS Communications Server: SNA Network Implementation*, SC31-8777
- *z/OS Communications Server: SNA Resource Definition Reference*, SC31-8778
- *Systems Network Architecture - Function Description of Logical Unit Types*, GC20-1868
- *Systems Network Architecture - Types of Logical Unit to Logical Unit Sessions*, GC20-1869.

### Advanced communications function for VTAM (ACF/VTAM)

- *Network Program Products General Information*, GC30-3350
- *Advanced Communications Function for VTAM Installation and Resource Definition*, SC23-0111
- *Advanced Communications Function for VTAM Customization*, SC23-0112
- *Advanced Communications Function for VTAM Operation*, SC23-0113
- *Advanced Communications Function for VTAM Messages and Codes*, SC23-0114
- *Advanced Communications Function for VTAM Diagnosis Guide*, SC23-0116
- *Advanced Communications Function for VTAM Diagnosis Reference*, LY30-5582

- *Advanced Communications Function for VTAM Data Areas*, LY30-5584
- *Advanced Communications Function for VTAM Programming*, SC23-0115
- *Advanced Communications Function for VTAM Reference Summary*, SC23-0135.

## **NetView Version 3.1**

- *NetView User's Guide*, SC31-8056
- *NetView Installation and Administration Guide*, SC31-8043
- *NetView Installation and Administration and Security Reference*, SC31-8045
- *NetView Customization Guide*, SC31-8052
- *NetView Customization: Writing Command Lists*, SC31-8055
- *NetView Automation Planning*, SC31-8051
- *NetView Automation Implementation*, SC31-8050
- *NetView RODM and GMFHS Programming Guide*, SC31-8049
- *NetView Messages*, SC31-8046

## **NetView MultiSystem Manager Version 2.2**

- *MultiSystem Manager: Open Topology Interface*, SC31-8144
- *MultiSystem Manager: Lovell NetWare Networks Open Topology Interface*, SC31-8129
- *MultiSystem Manager: OS/2 LAN Network Manager Networks*, SC31-8130
- *MultiSystem Manager: Internet Protocol Networks*, SC31-8131

## **DATABASE 2 (DB2)**

- *IBM DATABASE 2 Administration Guide*, SC26-4888
- *IBM DATABASE 2 Application Programming and SQL Guide*, SC26-4889
- *IBM DATABASE 2 Command and Utility Reference*, SC26-4891.

## **eNetwork Communications Server for OS/2 Warp, Version 5**

- *Quick Beginnings*, GC31-8189

## **Virtual Storage Access Method (VSAM)**

- *MVS/ESA Access Method Services Reference for VSAM Catalogs*, GC26-4075
- *MVS/ESA VSAM Administration Guide*, GC26-4151
- *MVS/ESA Catalog User's Guide*, GC26-4041.

## **Resource Access Control Facility (RACF)**

- *Resource Access Control Facility (RACF): General Information*, GC28-0722
- *System Programming Library: Resource Access Control Facility (RACF)*, SC28-1343
- *z/OS Security Server RACF Command Language Reference*, SA22-7687
- *Resource Access Control Facility (RACF) Security Administrators Guide*, SC28-1340.

## **System Modification Program Extended (SMP/E)**

- *System Modification Program Extended: User's Guide*, SC28-1302
- *System Modification Program Extended: (SMP/E) Terminal User's Guide*, SC28-1109

- *System Modification Program Extended: General Information*, GC28-1106
- *System Modification Program Extended: Reference*, SC28-1107.

## Sysplex planning

- *System/390 MVS Sysplex Application Migration*, GC28-1211

## DFSMS/MVS

- *z/OS DFSMSdfp Storage Administration Reference*, SC26-7402
- *DFSMS/MVS Access Method Services for ICF*, SC26-4906

## MVS

- *z/OS MVS Programming: Assembler Services Guide*, SA22-7605
- *z/OS MVS Programming: Assembler Services Reference, Volume 1*, SA22-7606
- *z/OS MVS Programming: Assembler Services Reference, Volume 2*, SA22-7607
- *z/OS MVS Setting Up a Sysplex*, SA22-7625.
- *z/OS MVS Using the Subsystem Interface*, SA22-7642.
- *z/OS MVS Installation Exits*, SA22-7593.
- *z/OS MVS Programming: Authorized Assembler Services Reference Vol 1*, SA22-7609
- *z/OS MVS Programming: Authorized Assembler Services Reference Vol 2*, SA22-7610
- *z/OS MVS Programming: Authorized Assembler Services Reference Vol 3*, SA22-7611
- *z/OS MVS Programming: Authorized Assembler Services Reference Vol 4*, SA22-7612
- *z/OS MVS Planning: Workload Management*, SA22-7602.
- *z/OS MVS Initialization and Tuning Guide*, SA22-7591
- *z/OS MVS Initialization and Tuning Reference*, SA22-7592
- *z/OS MVS Routing and Descriptor Codes*, SA22-7624
- *z/OS Program Directory*, GI10-6730
- *z/OS UNIX System Services Planning*, GA22-7800

---

## Determining if a publication is current

IBM regularly updates its publications with new and changed information. When first published, both hardcopy and BookManager softcopy versions of a publication are usually in step. However, due to the time required to print and distribute hardcopy books, the BookManager version is more likely to have had last-minute changes made to it before publication.

Subsequent updates will probably be available in softcopy before they are available in hardcopy. This means that at any time from the availability of a release, softcopy versions should be regarded as the most up-to-date.

For CICS Transaction Server books, these softcopy updates appear regularly on the *Transaction Processing and Data Collection Kit* CD-ROM, SK2T-0730-xx. Each reissue of the collection kit is indicated by an updated order number suffix (the -xx part). For example, collection kit SK2T-0730-06 is more up-to-date than SK2T-0730-05. The collection kit is also clearly dated on the cover.

Updates to the softcopy are clearly marked by revision codes (usually a # character) to the left of the changes.

---

## Accessibility

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully.

You can perform most tasks required to set up, run, and maintain your CICS system in one of these ways:

- using a 3270 emulator logged on to CICS
- using a 3270 emulator logged on to TSO
- using a 3270 emulator as an MVS system console

IBM Personal Communications provides 3270 emulation with accessibility features for people with disabilities. You can use this product to provide the accessibility features you need in your CICS system.





# Index

## Special characters

- supplied procedures, copying into a 62
- &CICS\_DIRECTORY symbol
  - z/OS UNIX subdirectory 236
- &JAVA\_HOME
  - Java subdirectory 236

## A

- ACB (VTAM), authorizing the region userid to open 82
- ACBNAME, parameter of VTAM APPL statement 118
- ACBNAME, VTAM APPL statement parameter
  - CMAS 125
- ACCEPT, function of SMP/E 60, 207
- accessibility options 300
- ACCTFILE, primer sample application accounts data set 223
- ACIXFILE, primer sample application index data set 223
- activating real-time analysis
  - VTAM definitions
    - for CMAS 126
- address space (CMAS)
  - add CICS system definitions 258
  - CICS SIT parameters 263, 305
  - create data repository 257
  - define VTAM requirements 124
    - activate the major nodes 126
    - define APPL statement 124, 125
    - define cross-domain resources 124, 125
    - define information to CICSplex 270
    - update configuration list 126
  - journalmodel considerations 229
  - preparing for API 95
  - restarting 274
  - START command for a CMAS 269
  - start ESSS 94
  - starting
    - as a batch job 255
    - from the system console 254
    - update IEASYSxx 101
- ADDTVOL, parameter of DFHISTAR 46
- ADFHAPD1 distribution library 36
- ADFHAPD2 distribution library 36
- ADFHC370 elements 36
- ADFHCLIB distribution library 36
- ADFHCOB elements 36
- ADFHINST distribution library 36
- ADFHMAC distribution library 36
- ADFHMLIB distribution library 36
- ADFHMOD distribution library 36
- ADFHMSGGS distribution library 36
- ADFHMSRC distribution library 36
- ADFHPARM distribution library 36
- ADFHPL1 elements 36
- ADFHPROC distribution library 36
- ADFHSAMP distribution library 36
- AINDEX, parameter of DFHISTAR 45
- alerts to NetView
  - preparing NetView to receive 254
- ALINDEX, parameter of DFHISTAR 45
- allocating
  - distribution and target libraries 58
  - RELFILE data sets 58
  - SMP/E data sets 58
- allocating libraries for IPCS 148
- APAR fixes 207, 208
- APF (authorized program facility) 79
- APF, MVS initialization parameter
  - IEASYSxx value 100
- APPC parameter, VTAM APPL statement 125
- APPL statement
  - CAS 125
  - CMAS 125
- application programming interface (API)
  - installing REXX function package 96
  - preparing for 95
- APPLID
  - accessing from several MVS images 119
  - notifying VTAM of the CICS APPLID 338
  - using for VTAM logon 341
- APPLY, function of SMP/E 60, 207
- applying service
  - supplied SMP/E procedure 211
- applying service to CICS 207
  - corequisite fixes 208
  - servicing the CICS messages data set 209
- ASMHCL exec, to create an MVS nucleus module list for CICS 116
- ASMPPTS, parameter of DFHISTAR 46
- ASMPMTS, parameter of DFHISTAR 45
- ASMPSCDS, parameter of DFHISTAR 45
- ASMPSTS, parameter of DFHISTAR 46
- ASREXIT 159
- ATCCONxx, VTAM list member 126
- ATCSTRxx, VTAM list member 126
- attributes for SMP/E data sets 39, 40
- AUTH, parameter of VTAM APPL statement 118
- AUTH, VTAM APPL statement parameter
  - CMAS 125
- authorized libraries 79
- authorized program facility (APF) 79
- authorizing hlq.SDFJAUTH 238
- authorizing SDFHAUTH 79
- authorizing SDFHLINK 79
- authorizing SDFHLPA 79
- authorizing SEYUAUTH 79
- authorizing SEYULINK 79
- authorizing SEYULPA 79
- auto-import options 294
- AUTOIMPORTDSN 294
- AUTOIMPORTMEM 294
- AUTOIMPORTTDQ 294
- automated-operation programs
  - message formatting coexistence considerations 108

- automatic installation of terminals 335
- AUTOREFRESH 297
- AUTORESETTIME 286
- auxiliary storage usage 253
- auxiliary trace (A) data set, DFHAUXT
  - creating for a CICS region 221
  - DD statement in IVP startup procedure 330
- auxiliary trace (B) data set, DFHBUXT
  - creating for a CICS region 221
  - DD statement in IVP startup procedure 330
- AUXTR 286
- AZONE, parameter of DFHISTAR 45
- AZONECSI, parameter of DFHISTAR 45
- AZONELOG, parameter of DFHISTAR 45

## B

- backing up the data repository 261
- BBACTDEF DD statement
  - with CMAS startup JCL 257
- BLKFB80 parameter of the DFHISTAR job 35
- BLKU parameter of the DFHISTAR job 35
- block sizes for data sets
  - block sizes for 35
- BLSCECT
  - updating 147
- browser connection 304

## C

- cache sets, VSAM RLS 162
- catalogs
  - distribution and target libraries 58
- category 1 transactions, authorizing the region
  - userid 83
- CBIPO method of installing 5
- CBPDO method of installing 5
  - description of 5
- CDRSC
  - CMAS-update configuration 126
  - create or access member
    - in SYS1.VTAMLST library 125, 126
  - defining 125
- CDRSC TI, parameter of VTAM START command 120
- CDS (control data set), saved 41
- CEDA transaction 343
  - installing the CICS sample programs 343
- CEECCICS, Language Environment interface
  - module 233
- CEEMSG, transient data destination, Language
  - Environment 233
- CEEOUT, transient data destination, Language
  - Environment 233
- CESE, transient data destination, Language
  - Environment 233
- CESO, transient data destination, Language
  - Environment 233
- checklists for setup
  - CMAS 9
  - MAS 9
  - overview 9

- checklists for setup (*continued*)
  - WUI 9
- CHS 287
- CHS1 287
- CHS2 287
- CICS Information Center
  - installing and configuring 73
- CICS libraries
  - authorizing 79
  - SMP/E function ACCEPT (DFHINST6) 60
- CICS Monitoring Facility (CMF) 281
- CICS SIT parameters 305
  - CMAS-related 263
  - MVS MAS-related 280
- CICS Sockets 244
- CICS storage in a CMAS 269
- CICS system definition (CSD) files
  - updating 227
- CICS system definition data set, DFHCSD 336
  - creating for all CICS regions 220
  - DD statement in IVP startup procedure 330
  - requirements for autoinstall 336
- CICS system initialization parameters 286
  - AUXTR 286
  - CPSMCONN 286
  - CWAKEY 286
  - INITPARM
    - EYU9VKEC 287
    - EYU9VWAN 287
  - SYSTR 286
  - USERTR 286
  - WRKAREA 286
- CICS TS
  - DASD storage requirements 8
  - installing 19, 25
  - introduction 3
  - required software and hardware 7
- CICS Web support 286
- CICS-supplied transactions in interactive IVP
  - from a terminal 341
- CICSPlex SM installation attributes 51
- CICSPlex SM libraries
  - authorizing 79
- CICSPLEX system parameter 308
- CMAC, CICS messages facility
  - DD statement in IVP startup procedure 329
  - defining resources needed 223
  - defining the messages data set, DFHCMACD 223
  - messages data set, DFHCMACD 329
  - resources needed to use with the IVP jobs 328
  - servicing the messages data set, DFHCMACD 209
- CMACVOL parameter of the DFHISTAR job 38
- CMAS
  - create data sets 267
  - customize data sets 267
  - setup worksheet 12
- CMAS ( address space)
  - add CICS system definitions 258
  - CICS SIT parameters 263, 305
  - define VTAM requirements 124
    - activate the major nodes 126

CMAS ( address space) *(continued)*  
   define VTAM requirements *(continued)*  
     define APPL statement 124, 125  
     define cross-domain resources 124, 125  
     update configuration list 126  
   journalmodel considerations 229  
   preparing for API 95  
   restarting 274  
   START command for a CMAS 269  
   start ESSS 94  
   update IEASYSxx 101  
 CMAS (address space)  
   create data repository 257  
   define VTAM requirements  
     define information to CICSplex 270  
   starting  
     as a batch job 255  
     from the system console 254  
 CMAS (CICSplex SM address space)  
   controlling CICS storage 269  
   controlling tasks 266  
 CMAS journaling 270  
 CMAS setup worksheet 12  
 CMAS-specific JCL requirements  
   BBACTDEF 257  
   CICS EXEC 255  
   DFHRPL 256  
   EYUCMASP 255  
   EYUDREP 256  
   EYULOG 256  
   EYUPARM 257  
   STEPLIB 256  
 CMASNAME parameter of EYU9XDUT 258  
 CMASNAME parameter of the DFHISTAR job 51  
 CMASYSID system parameter 308  
 CMD, MVS initialization parameter  
   IEASYSxx value 100  
 CMSSYSID parameter of the DFHISTAR job 51  
 COBOL under Language Environment 234  
 codepage 287  
   CHS1 287  
   CHS2 287  
   ENU1 287  
   JPN1 287  
 codepage conversion table 288  
 coexistence considerations  
   automated-operation programs 108  
   releases of CICS before this release 108  
 COIRTASKPRI system parameter 308  
 COLG 301  
 color of screens 300  
 COLORINK 300  
 COLORINKBANNER 300  
 COLORINKLINK 300  
 COLORINKVLINK 300  
 COLORPAPER 300  
 COLORPAPERALT 300  
 COLORPAPERERROR 300  
 COLORPAPERHEAVY 300  
 COLORPAPERLIGHT 300  
 COLORPAPERRULE 300  
 COLORPAPERWARN 300  
 COMMNDaa, operator commands member  
   for CMAS 254  
 common CICS data sets, creating (DFHCOMDS  
   job) 220  
 communications  
   CMAS-to-CMAS  
     defining cross-domain resources 125  
   configuration views  
     defining CMAS-to-CMAS links 270  
   console device for the IVP jobs  
     defining a TSO user 338  
     definition 335  
     using transactions 344  
 CONSOLE macro 167  
 console message formatting  
   activating 106  
   default initialization parameters 105  
   defining CICS as an MVS subsystem 99  
   modules needed 107  
 console messages 167  
 console support  
   defining a TSO user as a console 338  
   TSO CLIST processing 345  
 consolidated software inventory (CSI) 42  
 converting the data repository 260  
 COSD transaction 273  
 coupling facility 169  
   definition 178  
   recommendations 177  
   structures 178  
 coupling facility cache structures, VSAM RLS 161  
 coupling facility lock structure, VSAM RLS 161  
 COU, transient data extrapartition data set  
   DD statement in IVP startup procedure 330  
 COVA 303  
 COVC 303, 304  
 COVE 301, 303  
 COVG 303  
 COVI 301  
 COVP 301, 303  
 COVU 303  
 CPSM.EYUDREP  
   in CMAS startup JCL 256  
 CPSM.SEYUADEF  
   in CMAS startup JCL 255  
 CPSM.SEYUAUTH  
   in CMAS startup JCL 256  
 CPSM.SEYULOAD  
   in CMAS startup JCL 256  
 CPSM.SEYUVDEF  
   in CMAS startup JCL 255  
 CPSM310.EYUDREP  
   converting the data repository 260  
 CPSM310.SEYUAUTH  
   in MAS startup JCL 278  
 CPSM310.SEYULOAD  
   in MAS startup JCL 278  
 CPSM310.XEYUINST 211  
 CPSMCONN 286

- creating CICS system definition
  - data repository 257
- cross system locking, VSAM RLS 161
- cross-domain resources (CDRSC)
  - CMAS-update configuration 126
  - create or access member
    - in SYS1.VTAMLST library 125, 126
    - defining 125
- cross-system coupling facility (XCF)
  - used for interregion communication 240
- CSD files
  - updating 227
- CSI (consolidated software inventory) 42
- CSNE transient data destination 119
- CSSLIB parameter of the DFHISTAR job 46
- CSYSNAME parameter of the DFHISTAR job 51
- CSYSPLEX parameter of the DFHISTAR job 51
- CSYSYSID parameter of the DFHISTAR job 51
- customizable view and menu help 292
- customizable view and menu help data set 292
- customizing CICS
  - adding ISC support 241
  - installing CICS modules in MVS link pack area 129
  - using HPO support 115
- CVDASTYLE 295
- CWAKEY 286

## D

- DASD storage requirements
  - for CICS TS 8
- DASD-only logging
  - recommendations 195
- data repository
  - backing up 261
  - converting 260
  - creating 257
  - defining forward recovery log for 262
  - expanding 261
  - for each CMAS 257
  - identified in CMAS startup JCL 256
- data sets
  - RELFILE, allocating 58
- data sets (non-SMP/E), assigning a high-level index 44
- data sets for CICS 218
  - common to all regions, creating (DFHCOMDS job) 220
  - unique to each region, creating (DFHDEFDS job) 221
- DATEFORMAT 295
- DATESEPARATOR 296
- DAYLIGHT parameter of EYU9XDUT 258
- DBCTL
  - installation verification procedure (IVP) 351
- debugging profiles alternate index data set, DFHDPFMP
  - creating for each CICS region 222
- debugging profiles base data set, DFHDPFMB
  - creating for each CICS region 222
- debugging profiles path data set, DFHDPFMP
  - creating for each CICS region 222

- DECIMALSEPARATOR 296
- Default CICS userid, defining to RACF 91
- default EXCI pipe allocation limit 109
- DEFAULTCICSplex 298
- DEFAULTCICSRGN 298
- DEFAULTCMASCTXT 297
- DEFAULTCONNECT 298
- DEFAULTCONTEXT 297
- DEFAULTCSYSGRP 298
- DEFAULTDB2SS 298
- DEFAULTEJCOBEAN 298
- DEFAULTEJDJBEAN 298
- DEFAULTEVENT 298
- DEFAULTLOCFILE 298
- DEFAULTLOCTRAN 298
- DEFAULTMAPBAS 297
- DEFAULTMAPCOLL 297
- DEFAULTMAPMON 297
- DEFAULTMAPRTA 297
- DEFAULTMAPWLM 297
- DEFAULTMENU 297
- DEFAULTNAVIGATE 297
- DEFAULTPROGRAM 298
- DEFAULTTREMFILE 298
- DEFAULTTREMTRAN 299
- DEFAULTSCOPE 297
- DEFAULTTASK 299
- DEFAULTWARNCNT 297
- defining a terminal for online IVP 335
- defining IPIC connections 242
- Defining the hlq.SDFHLPA library to your MVS 133
- DEFVOL parameter of the DFHISTAR job 36
- DFH\$AFLA CSD group, FILEA sample programs 343
- DFH\$BTCH, input to the verify batch job 331
- DFH\$FILA CSD group, FILEA sample data set 343
- DFH\$SIP2 member of SYSIN data set 327
- DFH\$SIPn members of SYSIN data set, for IVP jobs 327
- DFH99SVC, SVC services
  - installing in the LPA 130
- DFHASMVS, -supplied procedure 62
- DFHAUPLE, -supplied procedure 62
- DFHAUXT data set, auxiliary trace
  - creating for a CICS region 221
  - DD statement in IVP startup procedure 330
- DFHBMSU, -supplied procedure 62
- DFHBUXT data set, auxiliary trace
  - creating for a CICS region 221
  - DD statement in IVP startup procedure 330
- DFHCMACD, messages data set 209, 223
  - defining associated CICS resources 223
- DFHCMACI, job to create and initialize the messages data set 223
- DFHCNV 288
- DFHCOMDS, job to create common CICS data sets 220
- DFHCSD data set, CICS system definition 336
  - creating for all CICS regions 220
  - DD statement in IVP startup procedure 330
  - requirements for autoinstall 336

DFHCSDUP requirements  
  DFHCSD 229  
  SYSIN 229  
DFHCSVC, CICS Type 3 SVC 130  
DFHCSVC, the CICS type 3 SVC  
  correct-level check 111  
  defining to MVS 111  
  installing 111  
  using for MRO between different CICS releases 113  
  using more than one version 112  
DFHCTAIX, sample application alternate index 223  
DFHCTCUS, sample application details data set 223  
DFHCTHLP, sample application help data set 223  
DFHCXRF, transient data extrapartition data set  
  DD statement in IVP startup procedure 330  
  local request queue data set 330  
DFHDEFDS, job to create data sets for each region 221  
DFHDLI user exits  
  XDLIPOST 232  
  XDLIPRE 232  
DFHDLPSB TYPE=ENTRY macro (remote DL/I) 231  
DFHDMPA data set, dump  
  creating for a CICS region 221  
  DD statement in IVP startup procedure 330  
DFHDMPB data set, dump  
  creating for a CICS region 221  
  DD statement in IVP startup procedure 330  
DFHDPFMB debugging profiles alternate index data set  
  creating for each CICS region 222  
DFHDPFMB debugging profiles base data set  
  creating for each CICS region 222  
DFHDPFMB debugging profiles path data set  
  creating for each CICS region 222  
DFHDSPEX, CICS post exit stub  
  installing in the LPA 130  
DFHDUMPX, CICS dump query exit  
  installing in the LPA 130  
DFHEBTAL, -supplied procedure 62  
DFHEITAL, -supplied procedure 62  
DFHEXTAL, -supplied procedure 62  
DFHGCD data set, global catalog  
  creating for a CICS region 221  
  DD statement in IVP startup procedure 330  
DFHHPSVC, VTAM authorized path SVC  
  defining to MVS 111, 115  
  loading into the MVS nucleus 115  
DFHHTML 292, 301, 303  
DFHHTML data set for HTML templates  
  creating for a CICS region 221  
DFHIJVMJ 61, 236  
DFHINST1, installation job 58  
DFHINST2, installation job 58  
DFHINST3, installation job 58  
DFHINST4, installation job 59  
DFHINST5, installation job 59  
DFHINST6, installation job 60  
DFHINSTA  
  submitting 66  
DFHINTRA data set, intapartition transient data  
  creating for a CICS region 221  
  DD statement in IVP startup procedure 330  
DFHIRP, interregion communication program  
  installing in the LPA 130  
  MRO requirement 240  
DFHISMKD, installation job 58  
DFHISTAR  
  CMASNAME parameter 258  
  customizing CMAS post-installation jobs 268  
  customizing MAS post-installation jobs 277  
  customizing WUI post-installation jobs 291  
  DSINFO 258  
  editing 65  
  post-installation members for a CMAS 267  
  post-installation members for a MAS 276  
  post-installation members for a WUI 290  
  submitting 65  
DFHISTAR, creating post-installation jobs 216  
DFHISTAR, job to define installation-related jobs 31  
  BLKFB80 parameter 35  
  BLKU parameter 35  
  CMACVOL parameter 38  
  CMASNAME parameter 51  
  CMSSYSID parameter 51  
  CSSLIB parameter 46  
  CSYSNAME parameter 51  
  CSYSPLEX parameter 51  
  CSYSYSID parameter 51  
  DEFVOL parameter 36  
  DINDEX parameter 33  
  DISTVOL parameter 36  
  DSINFO parameter 44  
  DZONE parameter 43  
  DZONECSI parameter 42  
  DZONELOG parameter 41  
  GINDEX parameter 33  
  GZONE parameter 43  
  GZONECSI parameter 42  
  GZONELOG parameter 41  
  JAVADIR parameter 48  
  JES parameter 32  
  JOB parameter 31  
  LIB parameter 31  
  LINDEX parameter 33  
  LOGGER-INFO parameter 50  
  NEWDREP parameter 51  
  OLDDREP parameter 52  
  OPTVOL parameter 37  
  parameter default values 26  
  PREFIX parameter 33  
  SCEEBND2 dsname 50  
  SCEECICS dsname 49  
  SCEECPP parameter 47  
  SCEELIB dsname 49  
  SCEELKED parameter 47  
  SCEELKEX parameter 47  
  SCEEOBJ parameter 47  
  SCEERUN dsname 49  
  SCEERUN2 dsname 49  
  SCEESAM parameter 47

DFHISTAR, job to define installation-related jobs  
(*continued*)

- SCLBSID parameter 47
- SCOPE parameter 32, 54
- SCSFMOD0 dsname 49
- SCSQAU parameter 48
- SCSQCI parameter 48
- SCSQL parameter 47
- SCSQLE parameter 48
- SEZACMTX dsname 49
- SEZARPCL dsname 49
- SISLOAD parameter 46
- SIXMEXP dsname 50
- SMP/E zone and zone log dispositions 43
- SMPLTS parameter 41
- SMPMTS parameter 40
- SMPPTS parameter 40
- SMPSCDS parameter 41
- SMPSTS parameter 40
- SMPVOL parameter 37
- SMPWORK parameter 40
- SMS parameter 36
- TAPEUNIT parameter 44
- TARGVOL parameter 37
- TCPIPHST parameter 52
- TCPIPPRT parameter 52
- TEMPLIB parameter 31
- TIMEZONE parameter 52
- TINDEX parameter 33
- TZONE parameter 43
- TZONECSI parameter 42
- TZONELOG parameter 41
- UTILITIES parameter 32
- WORKUNIT parameter 35
- WUI parameter 52
- WUINAME parameter 52
- WUIPLEX parameter 52
- WUISYSID parameter 53
- DFHIVPBT, batch IVP 323, 330
  - DFH\$BTCH input 331
  - expected output 333
  - system initialization parameters 327
- DFHIVPOL, interactive IVP 323, 335
  - CICS-supplied transactions 341
  - sample programs 344
  - terminating CICS 345
  - transactions 344
- DFHLCD data set, CICS local catalog
  - creating for each CICS region 221
  - DD statement in IVP startup procedure 330
- DFHLNKVS, -supplied procedure 62
- DFHMAPS, -supplied procedure 62
- DFHMAPT, -supplied procedure 62
- DFHRPL (dynamic transaction routing exit)
  - in CMAS startup JCL 256
  - in MAS startup JCL 278
- DFHSIP, CICS initialization program 335
- DFHSMPE, -supplied procedure 62
- DFHSMPE, cataloged procedure 207
- DFHSSEN, subsystem interface end-of-memory /  
end-of-task clean up routine 130
- DFHSSEN, subsystem interface end-of-memory /  
end-of-task clean up routine (*continued*)
  - installing in the LPA 130
- DFHSSGC, connect module
  - installing in the LPA 130
- DFHSSlly, initialization member 108
- DFHSSlly, message-formatting initialization  
member 104
- DFHSSWT, WTO router
  - installing in the LPA 130
- DFHSTART, -supplied procedure 62
- DFHSTART, CICS startup procedure 328
- DFHTEMP data set, temporary storage
  - creating for each CICS region 221
  - DD statement in IVP startup procedure 330
- DFHYBTPL, -supplied procedure 62
- DFHYBTVL, -supplied procedure 62
- DFHYITDL, -supplied procedure 62
- DFHYITEL, -supplied procedure 62
- DFHYITPL, -supplied procedure 62
- DFHYITVL, -supplied procedure 62
- DFHYXTDL, -supplied procedure 62
- DFHYXTEL, -supplied procedure 62
- DFHYXTPL, -supplied procedure 62
- DFHYXTVL, -supplied procedure 62
- DFHZITCL, -supplied procedure 62
- DFHZITDL, -supplied procedure 63
- DFHZITEL, -supplied procedure 63
- DFHZITFL, -supplied procedure 63
- DFHZITGL, -supplied procedure 63
- DFHZITPL, -supplied procedure 63
- dfjejbpl.policy 236
- DFLTUSER SIT parameter
  - for MVS MAS 280
- DINDEX parameter of the DFHISTAR job 33
- dispositions, zone and zone log 43
- distribution libraries
  - allocating and cataloging 58
  - SMP/E function ACCEPT (DFHINST6) 60
- distribution zone (DZONE) 43
  - CSI 42
  - log 41
- DISTVOL parameter of the DFHISTAR job 36
- DL/I
  - adding remote DL/I support 230
  - database control (DBCTL) 230
  - defining a PSB directory for remote DL/I  
support 231
  - DFHDLPSB TYPE=ENTRY (remote DL/I) 231
  - function shipping 230
  - global user exits 232
  - MXSSASZ parameter (remote DL/I) 231
  - requirements for remote database access 231
  - system initialization parameters (remote DL/I) 231
- DNS 244
- downward-compatible 130
- DSINFO parameter of the DFHISTAR job 44
- dump (A) data set, DFHDMPA
  - creating for a CICS region 221
  - DD statement in IVP startup procedure 330

- dump (B) data set, DFHDMPB
  - creating for a CICS region 221
  - DD statement in IVP startup procedure 330
- dynamic LU alias 120
  - LUAPFX parameter 118
- dynamic parse validation routines, installing 95
- DZONE parameter of the DFHISTAR job 43
- DZONECSI parameter of the DFHISTAR job 42
- DZONELOG parameter of the DFHISTAR job 41

## E

- EAS parameter, VTAM APPL statement 125
- EAS, parameter of VTAM APPL statement 118
- ECI 243
- ECLOGMSG system parameter 313
- elapsed time during installation of 56
- elements in CICS TS
  - list of 4
- ELPA (extended link pack area)
  - eligible modules 131
- ENU 287
- ENU1 287
- Environment Services System Services (ESSS)
  - adding to MVS linklist 94
  - description of 94
- ENVIRONMENT, EYUIPCSP EXEC parameter
  - IMBED statements required in BLSCECT 147
- ESSS (Environment Services System Services)
  - adding to MVS linklist 94
  - description of 94
- exception trace 286
- EXCI pipe allocation 108
- EXCI pipe allocation limit 109
- exclusive element or feature
  - description 4
- expanding the data repository 261
- extended recovery facility (XRF)
  - sessions, HAVAIL, parameter of VTAM APPL 118
- EYU\$CNV1 288
- EYU9VKEC 287
- EYU9VWAN 287
- EYU9XDUT 258
- EYU9XESS module 94
- EYUCHS1 288
- EYUCHS2 288
- EYUCMASJ 217
- EYUCMASP 217
- EYUCMASP, -supplied procedure 63
- EYUCMS01
  - CICS SIT parameters 264
- EYUCMS0P 217
- EYUCMS1A 297
- EYUCMSDS 217
- EYUCMSSP 217
- EYUCOVE 301, 303
- EYUCOVI 301, 303
- EYUCSYDS 218
- EYUCSYSJ 218
- EYUCSYSP 218
- EYUCSYSP, -supplied procedure 63

- EYUDREP data set
  - converting the data repository 260
  - creating the data repository 257
  - in CMAS startup JCL 256
  - storing CMAS-to-CMAS link definitions 270
- EYUENU1 288
- EYUHISTx 282
- EYUHISTx DD statement 279
- EYUJHIST 218
- EYUJPN1 288
- EYUJWREP 218, 289
- EYULMS0P 218
- EYULMSSP 218
- EYULOG 301, 303
- EYULOG DD statement
  - CMAS startup JCL 256
- EYULPMOD 218
- EYUPARM DD statement
  - in CMAS startup JCL 257
  - in MAS startup JCL 278
- EYUPARM parameters
  - description of 307
  - summary of 305
- EYUPLX01 297
- EYUWREP 303
- EYUWREP (Web User Interface server repository) 288, 301
- EYUWUI 301, 303
- EYUWUI0P 218
- EYUWUIDS 218
- EYUWUIIN 218
- EYUWUIJ 218
- EYUWUIP 218
- EYUWUIP, -supplied procedure 63
- EYUWUISP 218

## F

- features in CICS TS
  - list of 4
- FILEA data set, sample program data
  - creating for each CICS region 222
  - DD statement in IVP startup procedure 330
- FILEA sample applications
  - transactions used 325
- FORMATMSG, parameter of DFHSSIyy 105
- forward recovery log for data repository 262
- function shipping 230

## G

- generic routecodes 105
- gethostbyaddr function 244
- GID 86
- GIMMPDFT modules, attributes of SMP/E data sets 39
- GIMSP messages and control statements 60
- GIMZPOOL, member of SYS1.MACLIB 59
- GINDEX parameter of the DFHISTAR job 33
- global catalog data set, DFHGCD
  - creating for a CICS region 221
  - DD statement in IVP startup procedure 330

global zone (GZONE) 43  
 CSI 42  
 log 41  
 GLOBALPREFILTER 298  
 GMMTEXTMSG 296  
 group identifier (GID) 86  
 GRPLIST  
 MVS MAS parameter 280  
 GZONE parameter of the DFHISTAR job 43  
 GZONECSI parameter of the DFHISTAR job 42  
 GZONELOG parameter of the DFHISTAR job 41

## H

hardware requirements  
 installing CICS TS 7  
 HAVAIL, parameter of VTAM APPL statement 118  
 help  
 customizable view and menu help 292  
 HFS0DSN parameter of the DFHISTAR job 34  
 HFS1DSN parameter of the DFHISTAR job 34  
 HFSADSN parameter of the DFHISTAR job  
 HFSADSN parameter 35  
 HIDEPASSWORD, parameter of DFHSSlyy 105  
 high-performance option (HPO) 111, 115  
 defining the CICS SVCs to MVS 111  
 DFHHPSVC, VTAM authorized path SVC 115  
 system initialization parameter 115  
 VTAM authorized path, DFHHPSVC 115  
 history recording 282  
 HISTRECSMSG 282  
 hostname 287, 304  
 HTML template data set, DFHHTML  
 creating for a CICS region 221  
 http 287, 304  
 HTTP 243

## I

ICCFCC, -supplied procedure 63  
 ICCFCCL, -supplied procedure 63  
 ICCFCGL, -supplied procedure 63  
 ICCFCL, -supplied procedure 63  
 IEASYSxx, MVS initialization member 99  
 establishing requirements 101  
 noting values in 100  
 updating  
 for a CMAS 101  
 IEFSSNaa, MVS subsystem initialization member 101  
 IEW0461 message from the binder 60  
 IEW2646 message from the binder 60  
 IEW2651 message from the binder 60  
 IIOF 243  
 implementing MVS workload management 151  
 import options 294  
 IMS, database control (DBCTL) 230  
 INACTIVETIMEOUT 299  
 initialize CMAS 255  
 initialize data repository 258  
 initialize MVS MAS 276, 278  
 INITPARM 287

INITPARM (*continued*)  
 EYU9VKEC 287  
 EYU9VWAN 287  
 installation  
 of CICS sample programs 343  
 installation jobs  
 elapsed time 56  
 messages 60  
 restarting 56  
 running 56  
 installation verification procedure (IVP) 323  
 applids, defining and activating 324  
 auxiliary temporary storage data set,  
 DFHTEMP 330  
 auxiliary trace data set (A), DFHAUXT 330  
 auxiliary trace data set (B), DFHBUXT 330  
 batch IVP, DFHIVPBT 330  
 CICS region definition data set 330  
 CICS startup procedure, DFHSTART 328  
 CMAC resources 328  
 data sets for 324  
 FILEA data set, sample application data 330  
 global catalog data set, DFHGCD 330  
 interactive IVP, DFHIVPOL 335  
 defining a terminal for 335  
 local catalog data set, DFHLCD 330  
 messages data set, DFHCMACD 329  
 RACF userid authority 324  
 required SVC for 324  
 security considerations 325  
 SYSIN data set 329  
 system definition data set 330  
 system initialization parameters 327  
 transaction dump (A) data set, DFHDMPA 330  
 transaction dump (B) data set, DFHDMPB 330  
 transient data extrapartition data set, COUT 330  
 transient data extrapartition data set,  
 DFHCXRF 330  
 transient data extrapartition data set, DFHLRQ 330  
 transient data extrapartition data set, LOGUSR 330  
 transient data extrapartition data set, MSGUSR 330  
 transient data extrapartition data set, PLIMSG 330  
 transient data intrapartition data set,  
 DFHINTRA 330  
 using the CEDA transaction 343  
 using transactions from console device 344  
 installing CICSplex modules in MVS link pack  
 area 141  
 eligible modules 141  
 installing Language Environment support 233  
 Installing modules in the MVS link pack area 133, 134  
 installing support for programming languages 232  
 installing system  
 software requirements 7  
 InstallShield  
 installing CICS Information Center 73  
 interface modules  
 CEECICS 233  
 Language Environment 233  
 interregion communication (IRC) 239



- intrapartition transient data data set, DFHINTRA
  - creating for a CICS region 221
  - DD statement in IVP startup procedure 330
- IP interconnectivity (IPIC)
  - defining connections 242
- IPCS tools
  - preparing to use 147
  - updating BLSCECT 147
  - updating library allocations 148
- IPCSLIB data set for IPCS 148
- IPIC
  - TCPIP initialization parameter 242
- ISC 243
- ISMF control data set application, VSAM RLS 162
- ISMF storage class application 162
- ISPMLIB data set for IPCS 148
- ISPPLIB data set for IPCS 148

## J

- Java
  - components 236
  - JVM 236
  - samples 237
  - verifying installation 236
  - z/OS UNIX files 237
- JAVADIR 61, 236
- JAVADIR parameter of the DFHISTAR job 48
- JES parameter of the DFHISTAR job 32
- JOB parameter of the DFHISTAR job 31
- journal records, CMAS 270
- journalmodels, CMAS 229
- JPN 287
- JPN1 287
- JRNLEFCH system parameter 309
- JRNLOPACT system parameter 310
- JRNLRTAEV system parameter 310
- JVM
  - JVM profiles 236
  - properties files 236
  - SYS1.PARMLIB(BPXPRMxx) parameters 238

## L

- language 287
- Language Environment 233
  - support for C language 234
  - support for COBOL 234
  - support for PL/I 235
- Language identifiers 287
  - CHS 287
  - ENU 287
  - JPN 287
- language support 285
- LIB parameter of the DFHISTAR job 31
- libraries
  - authorizing 79
  - creating multiple sets 64
  - load libraries
    - support for secondary extents 207
  - RACF-protecting 81

- libraries (*continued*)
  - SMP/E function ACCEPT (DFHINST6) 60
  - target libraries 37
- libraries, CICSplex SM
  - DFHISTAR related 217
  - skeleton jobs 217
- LINDEX parameter of the DFHISTAR job 33
- link editing modules
  - IEW0461 message 60
  - IEW2646 message 60
  - IEW2651 message 60
  - when running installation jobs 60
- link pack area (LPA)
  - CICS modules required in the LPA
    - DFH99SVC 130
    - DFHCSVC 130
    - DFHDUMPX 130
    - DFHIRP 130
    - DFHSPEX 130
    - DFHSSEN 130
    - DFHSSGC 130
    - DFHSSWT 130
  - eligible CICSplex modules 141
  - eligible modules 131
  - installing CICS modules in 129, 134
  - installing CICSplex modules in 141
  - installing required modules 130
  - space for CICS requirements 133
- link-editing
  - DFHHPSVC into MVS nucleus 115
  - messages during 60
- linkedit temporary store (LTS) 41
- linklist, installing CICS-required modules 95
- listener 243
- LNK, MVS initialization parameter
  - IEASYSxx value 100
- LNKAUTH, MVS initialization parameter
  - IEASYSxx value 100
- load libraries 79
  - secondary extents 207
- local catalog (DFHLCD)
  - creating for each CICS region 221
  - DD statement in IVP startup procedure 330
- local MAS 284
- log manager
  - coupling facility considerations 177
  - defining logger environment 169
  - requirements planning 169
  - using DASD-only logging 195
- log streams, authorizing the region userid to use 84
- logger environment
  - coupling facility space required for RLS AORs 193
  - RLS implications 193
  - staging data sets 194
- LOGGER-INFO parameter of the DFHISTAR job 50
- LOGMODE tables 118
  - required entries for autoinstall 336
- LOGMODE, parameter of VTAM APPL statement 118
- LOGUSR, transient data extrapartition data set
  - DD statement in IVP startup procedure 330

- LPA (link pack area)
  - CICS modules required in the LPA
    - DFH99SVC 130
    - DFHCSVC 130
    - DFHDUMPX 130
    - DFHIRP 130
    - DFHSPEX 130
    - DFHSSEN 130
    - DFHSSGC 130
    - DFHSSWT 130
  - eligible CICSplex modules 141
  - eligible modules 131
  - installing CICS modules in 129, 134
  - installing CICSplex modules in 141
  - installing required modules 130
  - space for CICS requirements 133
- LPA, MVS initialization parameter
  - IEASYSxx value 100
- LPA, system initialization parameter 138
- LTS (linkedit temporary store) 41
- LUAPFX, parameter of VTAM APPL statement 118
  - choosing a value 122
- LUTYPE 6 sessions
  - ACQ 118
  - PARSESS 118
  - PERSIST 118

## M

- macro temporary store (MTS) 40
- macros
  - DFHDLPSB TYPE=ENTRY (remote DL/I) 231
- map function
  - default map objects 297
  - WUI server initialization parameters 297
- MAS 284
  - create data sets 276
  - customize data sets 276
  - setup worksheet 13
- MAS setup worksheet 13
- MAS-specific JCL requirements
  - DFHRPL DD 278
  - EYUPARM DD 278
  - STEPLIB DD 278
- MASALTLRTCNT system parameter 310
- MASALTLRTPRI system parameter 310
- MASALTLRTTIM system parameter 310
- MASINITTIME system parameter 310
- MASPLTWAIT system parameter 310
- MAXCAD, MVS initialization parameter
  - IEASYSxx value 100
  - setting 101
- MAXPROCUSER 86
- MAXUSER, MVS initialization parameter
  - IEASYSxx value 100
- MAXUSERS 299
- message log (EYULOG)
  - with CMAS JCL 256
- messages
  - GIMSP 60
  - IEW0461, from the binder 60

- messages (*continued*)
  - IEW2646, from the binder 60
  - IEW2651, from the binder 60
  - SMP/E 60
    - when running installation jobs 60
- MODIFY command 344
- modules eligible for the LPA/ELPA 131
- modules, installing in LPA 129
- MRO between different CICS release levels 113
- MSGCASE 296
- MSGCASE system parameter 312
- MSGUSR, transient data extrapartition data set
  - DD statement in IVP startup procedure 330
- MTS (Macro temporary store) 40
- multiple libraries
  - creating 64
- multiregion operation (MRO)
  - between different CICS release levels 113
  - defining CICS as an z/OS subsystem 240
  - installation considerations 240
- MVS ASREXIT 159
- MVS definitions
  - CICS protection key 149
  - for CICS performance 151
  - for console messages 167
  - password checking 149
- MVS linklist
  - installing CICS-required modules 95
  - installing IGWABWO 97
  - installing IGWARLS 97
  - other MVS product modules 97
- MVS logger 169
- MVS subsystem, defining CICS as 99
- MVS workload management 151

## N

- NAME system parameter 312
- national language support 285
- NetView
  - sending generic alerts to 254
- NEWDREP parameter of the DFHISTAR job 51
- non-exclusive element or feature
  - description 4
- NQNMODE, parameter of VTAM START
  - command 120
- NSINTERADDR 244
- NSYSLX, MVS initialization parameter
  - IEASYSxx value 100
  - setting 101

## O

- OLDDREP parameter of the DFHISTAR job 52
- optional source material
  - optional source material 66
- optional source material, loading 66
- OPTVOL parameter of the DFHISTAR job 37

## P

parameters  
  CICS SIT  
    for CMAS 263  
    for MVS MAS 280  
  CICSplex SM system 305  
  in CMAS startup JCL 305  
  in MAS startup JCL 305  
PARSESS parameter, VTAM APPL statement  
  CMAS 125  
PARSESS, parameter of VTAM APPL statement 118  
PASS option of VTAM APPL parameter AUTH 125  
password checking 149  
PDIR (PSB directory) 230  
PDSE 8  
performance definitions for MVS 151  
performance parameters (CICS), matching to service policies 152  
PERSIST, parameter of VTAM APPL statement 118  
PLIMSG, transient data extrapartition data set  
  DD statement in IVP startup procedure 330  
port 287, 304  
post-installation jobs  
  EYUCMASP 255  
post-installation member 217  
post-installation members  
  EYUCMASJ 217  
  EYUCMASP 217  
  EYUCMS0P 217  
  EYUCMSDS 217  
  EYUCMSSP 217  
  EYUCSYDS 218  
  EYUCSYSJ 218  
  EYUCSYSP 218  
  EYUJHIST 218  
  EYUJWREP 218  
  EYULMS0P 218  
  EYULMSSP 218  
  EYULPMOD 218  
  EYUWUI0P 218  
  EYUWUIDS 218  
  EYUWUIIN 218  
  EYUWUIJ 218  
  EYUWUIP 218  
  EYUWUISP 218  
predefined LU alias 121  
PREFIX parameter of the DFHISTAR job 33  
problem state 79  
procedures, copying into a procedure library 62  
PROG, MVS initialization parameter  
  IEASYSxx value 100  
program temporary fix (PTF) 208  
properties files  
  samples 236  
protecting CICS load libraries 81  
protection key definition 149  
PRTAUXT step of IVP jobs 329  
PRTBUXT step of IVP jobs 329  
PRTDMPA step of IVP jobs 329  
PRTDMPB step of IVP jobs 329  
PRVMOD, system initialization parameter 138

PTF (program temporary fix) 207, 208  
PTF temporary store (PTS) 40  
PTS (PTF temporary store) 40

## R

RACF (resource access control facility)  
  activating resource classes 89  
  authorizing the region userid for category 1 transactions 83  
  authorizing the region userid to open a region's VTAM ACB 82  
  authorizing the region userid to use MVS log streams 84  
  defining the default CICS userid 91  
  installing the interface routines 95  
  IVP userid authority 324  
  LOGSTRM general resource class 84  
  protecting CICS load libraries 81  
RECEIVE, function of SMP/E 207  
references, unresolved 60  
region userid, authorizing for category 1 transactions 83  
region userid, authorizing to OPEN a region's ACB 82  
release synchronization of CMAS and CICS 253  
RELFILE data sets  
  allocating 58  
requirements for CICS TS  
  hardware 7  
  software 7  
resource classes, activating 89  
resource definitions  
  dynamic creation during CICSplex SM initialization 227  
  resource definitions, autoinstall for terminals 335  
RESOURCELIMIT 298  
RESSTATUS system parameter 312  
restarting CMAS 274  
restarting MVS MAS 283  
RESTORE, function of SMP/E 207  
REXX function package, installing 96  
routecodes, generic 105  
ROUTECODES, parameter of DFHSSlly 105  
RSVNONR, MVS initialization parameter  
  IEASYSxx value 100  
RSVSTRT, MVS initialization parameter  
  IEASYSxx value 100  
run times of installation jobs 56

## S

sample applications data sets, creating  
  ACCTFILE, primer sample application accounts 223  
  ACIXFILE, primer sample application index 223  
  DFHCTAIX, sample application alternate index 223  
  DFHCTCUS, sample application details 223  
  DFHCTHLP, sample application help 223  
sample JCL  
  CMAS 255  
  MAS 278

- sample JCL (*continued*)
  - to modify CICSplex SM supplied CICS resource definitions 228
- sample program data, FILEA
  - creating for each CICS region 222
  - DD statement in IVP startup procedure 330
- sample programs
  - EYU\$CNV1 288
  - EYUJWREP 289
- sample programs in interactive IVP 344
- sample programs, installing 343
- saved control data set (SCDS) 41
- SCDS (saved control data set) 41
- SCEEBND2 parameter of the DFHISTAR job 50
- SCEECICS parameter of the DFHISTAR job 49
- SCEECPP parameter of the DFHISTAR job 47
- SCEELIB parameter of the DFHISTAR job 49
- SCEELKED parameter of the DFHISTAR job 47
- SCEELKEX parameter of the DFHISTAR job 47
- SCEEOBJ parameter of the DFHISTAR job 47
- SCEERUN parameter of the DFHISTAR job 49
- SCEERUN2 parameter of the DFHISTAR job 49
- SCEESAMP parameter of the DFHISTAR job 47
- SCLBSID parameter of the DFHISTAR job 47
- SCOPE parameter of the DFHISTAR job 32, 54
- SCSFMOD0 parameter of the DFHISTAR job 49
- SCSQANLE parameter of the DFHISTAR job 48
- SCSQAUTH parameter of the DFHISTAR job 48
- SCSQCICS parameter of the DFHISTAR job 48
- SCSQLOAD parameter of the DFHISTAR job 47
- SDFHENV 61, 236
- SDFHPL1 target library 36
- SDFJAUTH 238
- SDSNLOAD parameter
  - SDSNLOAD parameter 49
- SDSNLOAD parameter of the DFHISTAR job 49
- SDUMP options 148
- SEC system parameter 313
- secondary extents, load libraries 207
- SECTIMEOUT system parameter 314
- secure sockets layer (SSL) 303
- security 302
  - COVA 303
  - COVC 303
  - COVE 303
  - COVG 303
  - COVP 303
  - COVU 303
  - DFHHTML 303
  - EYUCOVE 303
  - EYUCOVI 303
  - EYULOG 303
  - EYUWREP 303
  - EYUWUI 303
- security policy
  - dfjejbpl.policy 236
- security with the CICS IVPs 325
- SELECT, parameter of DFHISTAR 54
- server 284
- service, applying
  - supplied SMP/E procedure 211
- servicing CICS 207
  - corequisite fixes 208
- session outage notification (SON)
  - SONSCIP 118
- setting up
  - CMAS 226, 252
  - MVS MAS 274
  - SEZACMTX parameter of the DFHISTAR job 49
  - SEZARPCL parameter of the DFHISTAR job 49
- sharing control data sets, VSAM RLS 163
- Shutdown button 273
- shutting down a CMAS 273
- SIGNONPANEL 299
- SISLOAD parameter of the DFHISTAR job 46
- SIT parameters, CICS 305
  - CMAS-related 263
  - MVS MAS-related 280
- SIXMEXP parameter of the DFHISTAR job 50
- SMF, MVS initialization parameter
  - IEASYSxx value 100
- SMP/E data sets, attributes 39, 40
- SMP/E options, specifying 43
- SMP/E zone and zone log dispositions 43
- SMP/E, using to apply service 211
- SMP0001, SMP/E sample USERMOD for SMP/E data sets 39
- SMPCNTL, DD statement of DFHSMPE 208
- SMPLTS parameter of the DFHISTAR job 41
- SMPMTS parameter of the DFHISTAR job 40
- SMPPTS parameter of the DFHISTAR job 40
- SMPSCDS parameter of the DFHISTAR job 41
- SMPSTS parameter of the DFHISTAR job 40
- SMPVOL parameter of the DFHISTAR job 37
- SMPWORK parameter of the DFHISTAR job 40
- SMS parameter of the DFHISTAR job 36
- software requirements 7
  - installing system 7
- SONSCIP parameter, VTAM APPL statement 125
- SONSCIP, parameter of VTAM APPL statement 118
- source temporary store (STS) 40
- SPOOLCLASS system parameter 314
- SSI definitions for MRO 240
- SSL 244
- SSL (secure sockets layer) 303
- staging data sets 194
- STALL system parameters 314
- START command
  - CMAS 269
- starting
  - CMAS
    - as a batch job 255
    - from the system console 254
    - sample procedure 255
  - MVS MAS 278
- stopping
  - MVS MAS 283
- storage key definition 149
- STS (source temporary store) 40
- subsystem definition to z/OS 240
- subsystem, MVS, defining CICS as 99
- supervisor state 79

SUPPRESSCMF system parameter 281, 317  
 suppressing CMF records 281, 317  
 SVC  
   correct-level check for DFHCSVC 111  
   defining to MVS 111  
   DFHHPSVC, defining 115  
   DFHHPSVC, loading into the MVS nucleus 115  
   installing the CICS type 3 SVC 111  
   using DFHCSVC for MRO between different CICS releases 113  
   using more than one version of DFHCSVC 112  
 SVC Parm statement in an IEASVCcc member 111  
 SVCs, installing DFHCSVC in the LPA 130  
 SYMREC Authorization Exit 159  
 SYMREC macro 159  
 synchronization of CMAS and CICS releases 253  
 SYS1.CPSM320.SEYULPA  
   defining to your MVS 142, 143  
 SYS1.PARMLIB library  
   BLSCECT exit control table 147  
   COMMNDaa member with operator commands 254  
   IEASYSxx member  
     establishing requirements 101  
     when creating CMAS started tasks 101  
 SYS1.PROCLIB  
   CMAS startup procedure 254  
 SYS1.VTAMLST library  
   application definitions  
     CMAS 125  
   cross-domain resource definition 125  
 SYSID parameter of EYU9XDUT 258  
 SYSIN data set, partitioned  
   creating for all CICS regions 220  
   DD statement in IVP startup procedure 329  
 SYSMOD (system modification) 207, 211  
 sysplex, z/OS  
   cross-system coupling facility (XCF)  
     for MRO links across z/OS images 240  
 SYSPROC data set for IPCS 148  
 SYSTCPD 244  
 system abend 722 60  
 system initialization parameters 327  
   TCPIP 242  
 system modification (SYSMOD) 211  
 System Modification Program Extended (SMP/E)  
   ACCEPT function 60, 207  
   APPLY function 60, 207  
   DFHSMPE 207  
   messages 61  
   messages during link-editing 60  
   RECEIVE function 207  
   RESTORE function 207  
 system parameters  
   description of 307  
   summary of 305  
 SYSTR 286

**T**  
 tailoring CICS  
   installing MRO support 239

TAPEUNIT parameter of the DFHISTAR job 44  
 target libraries  
   allocating and cataloging 58  
   SMP/E function ACCEPT (DFHINST6) 60  
 target zone (TZONE) 43  
   CSI 42  
   log 41  
 TARGVOL parameter of the DFHISTAR job 37  
 TCP/IP  
   DNS 244  
   ECI 243  
   enablement 243  
   HTTP 243  
   IIOIP 243  
   installation requirements 244  
   ISC 243  
   listener 243  
   SSL 244  
 TCPIP, system initialization parameter 242  
 TCPIPADDRESS 293  
 TCPIPHOSTNAME 287, 293, 304  
 TCPIPHST parameter of the DFHISTAR job 52  
 TCPIPHTTPOST 293  
 TCPIPPOST 287, 293, 304  
 TCPIPVRT parameter of the DFHISTAR job 52  
 TCPIPSSL 294  
 TCPIPSSLCERT 294  
 TCPIPSSLCIPHERS 294  
 TEMPLIB parameter of the DFHISTAR job 31  
 temporary storage data set, DFHTEMP  
   creating for each CICS region 221  
   DD statement in IVP startup procedure 330  
 terminals  
   defining for the online IVP 335  
 THOUSNDSEPARATOR 296  
 times taken to run the installation jobs 56  
 TIMESEPARATOR 296  
 TIMEZONE parameter  
   EYUINST EXEC 258  
 TIMEZONE parameter of the DFHISTAR job 52  
 TINDEX parameter of the DFHISTAR job 33  
 TOBATCHREQ  
   CICSplex SM system parameter 317  
 TOONLINEREQ  
   CICSplex SM system parameter 317  
 TOPOLLINT  
   CICSplex SM system parameter 318  
 transaction dump data sets  
   dump (A) data set, DFHDMPA 330  
   dump (B) data set, DFHDMPB 330  
 transient data destination CSNE 119  
 transient data extrapartition data sets  
   destination before transient data initialization,  
     DFHCXRF 330  
   destination before transient data initialization,  
     DFHLRQ 330  
   destination used by C/370 application  
     programs 330  
   destination used by CICS services 330  
   destination used by PL/I application programs 330

- transient data extrapartition data sets *(continued)*
  - destination, LOGA, used by CICS sample programs 330
- transient data queue definitions 301
  - COLG 301
  - COVE 301
  - COVI 301
  - COVP 301
- TSO
  - defining a TSO user as a console 338
- type 3 SVC routine
  - used for interregion communication 239
- TZONE parameter of the DFHISTAR job 43
- TZONECSI parameter of the DFHISTAR job 42
- TZONELOG parameter of the DFHISTAR job 41

## U

- UID 86
- UNIX System Services 237
- UNIX System Services access
  - MAXPROCUSER 86
- unresolved references 60
- UPAGE option of VTAM APPL parameter AUTH 125
- updating CICSplex definition
  - BLSCECT 147
  - CSD files using DFHCSDUP 227
  - library allocations for IPCS
    - IPCSLIB 148
    - ISPMLIB 148
    - ISPPLIB 148
    - SYSPROC 148
- updating configuration list 126
- URL 287, 304
- user identifier (UID) 86
- USERTR 286
- USSDIR 61, 236
- USSDIRA, parameter of DFHISTAR 46
- UTILITIES parameter of the DFHISTAR job 32

## V

- VARY NET command 127
- verification tests, running IVP jobs 323
- verify batch job, DFHIVPBT 330
  - DFH\$BTCH input 331
  - expected output 333
- verify interactive job, DFHIVPOL 335
  - DFHIVPOL 341
  - invoking and executing sample programs 344
  - terminating CICS 345
  - using CICS-supplied terminal transactions 341
  - using transactions 344
- VSPACE option of VTAM APPL parameter AUTH 118
- VPACING parameter, VTAM APPL statement 125
- VPACING, parameter of VTAM APPL statement 119
- VSAM RLS
  - activating the coupling facility structures 165
  - adding new parameters to SYS1.PARMLIB 164
  - altering data set attributes for RLS access 162

- VSAM RLS *(continued)*
  - amending parameters in
    - SYS1.PARMLIB(BPXPRMxx) 102
  - authorizing CICS userids 164
  - CFRM policy 161
  - defining cache sets 162
  - defining coupling facility cache structures 161
  - defining sharing control data sets 163
  - defining SMS storage classes for RLS access 162
  - defining the coupling facility lock structure 161
  - definitions, required 161
  - establishing new procedures 164
  - ISMF control data set application 162
  - ISMF storage class application 162
  - IXCMIAPU utility 161

## VTAM

- APPL statement 117
  - ACBNAME parameter 118
  - AUTH parameter 118
  - cross-domain considerations 119
  - EAS parameter 118
  - HAVAIL parameter 118
  - LOGMODE parameter 118
  - LUAPFX parameter 118
  - PARSESS parameter 118
  - PERSIST parameter 118
  - SONSCIP parameter 118
  - VPACING parameter 119
- authorized path 115
- CICS APPLID 118
  - cross-domain considerations 119
  - definitions required for CICS 117
  - LOGMODE table entries for autoinstall 336
  - version, release, and modification level indicator 119
- VTAM ACB, authorizing the region userid to open 82
- VTAM LU alias 120
- VTAM requirements
  - activating definitions 126
  - CMAS 125
  - defining APPL statement 125
  - defining cross-domain resource 125
  - updating configuration list 126

## W

- Web User Interface server 284
  - create data sets 290
  - customize data sets 290
  - temporary storage models 301
- Web User Interface Server
  - setup worksheet 13
- Web User Interface server initialization optional parameters 293
- Web User Interface server initialization parameters 292
  - AUTOREFRESH 297
  - COLORINK 300
  - COLORINKBANNER 300
  - COLORINKLINK 300
  - COLORINKVLINK 300

Web User Interface server initialization parameters  
(continued)

COLORPAPER 300  
COLORPAPERALT 300  
COLORPAPERERROR 300  
COLORPAPERHEAVY 300  
COLORPAPERLIGHT 300  
COLORPAPERRULE 300  
COLORPAPERWARN 300  
CVDASTYLE 295  
DATEFORMAT 295  
DATESEPARATOR 296  
DECIMALSEPARATOR 296  
DEFAULTCICSplex 298  
DEFAULTCICSrgn 298  
DEFAULTCMASctxt 297  
DEFAULTCONNECT 298  
DEFAULTCONTEXT 297  
DEFAULTCSysgrp 298  
DEFAULTDB2SS 298  
DEFAULTEJCObean 298  
DEFAULTEJDJbean 298  
DEFAULTEVENT 298  
DEFAULTLOCFILE 298  
DEFAULTLOCTRAN 298  
DEFAULTMAPBAS 297  
DEFAULTMAPCOLL 297  
DEFAULTMAPMON 297  
DEFAULTMAPRTA 297  
DEFAULTMAPWLM 297  
DEFAULTMENU 297  
DEFAULTNAVIGATE 297  
DEFAULTPROGRAM 298  
DEFAULTREMFIL 298  
DEFAULTREMTAN 299  
DEFAULTSCOPE 297  
DEFAULTTASK 299  
DEFAULTWARNCNT 297  
GLOBALPREFILTER 298  
GMMTEXTMSG 296  
INACTIVETIMEOUT 299  
MAXUSERS 299  
MSGCASE 296  
RESOURCELIMIT 298  
SIGNONPANEL 299  
TCPIPADDRESS 293  
TCPIPHOSTNAME 287, 293, 304  
TCPIPHTTPHOST 293  
TCPIPSPORT 287, 293, 304  
TCPIPSSL 294  
TCPIPSSLCERT 294  
TCPIPSSLCIPHERS 294  
THOUSNDSEPARATOR 296  
TIMESEPARATOR 296  
WUITRACE 300

Web User Interface server initialization required parameters 292

Web User Interface server repository  
(EYUWREP) 288, 301

WLMLOADCOUNT system parameter 318  
WLMLOADTHRSH system parameter 318

workload management, MVS 151  
worksheet for IVP  
  CMAS 14  
  MAS 14  
  WUI 14  
worksheets for setup  
  CMAS 9  
  MAS 9  
  overview 9  
  WUI 9  
WORKUNIT parameter of the DFHISTAR job 35  
WRKAREA 286  
WTO (write-to-operator) macro 167  
WUI  
  importing data repository definitions 294  
  WUI parameter of EYU9XDUT 259  
  WUI parameter of the DFHISTAR job 52  
  WUI setup worksheet 13  
  WUIAPPLID parameter of EYU9XDUT 259  
  WUINAME parameter of EYU9XDUT 259  
  WUINAME parameter of the DFHISTAR job 52  
  WUIPLEX parameter of EYU9XDUT 259  
  WUIPLEX parameter of the DFHISTAR job 52  
  WUISYSID parameter of EYU9XDUT 259  
  WUISYSID parameter of the DFHISTAR job 53  
  WUITRACE 300

## X

XDLIPOST global user exit 232  
XDLIPRE global user exit 232  
XRF (extended recovery facility)  
  sessions, HAVAIL, parameter of VTAM APPL 118  
XTRAQUAL parameter  
  XTRAQUAL parameter 35  
XTRAQUAL parameter of the DFHISTAR job 35  
XXRSTAT exit, effect on takeover of a VTAM ACB 82

## Z

z/OS image  
  XCF/MRO is required for links between CICS regions  
  in different 240  
z/OS UNIX 8, 237  
z/OS UNIX access 86  
z/OS UNIX System Services 237  
z/OS UNIX System Services access 86  
zone and zone log dispositions 43  
ZONEOFFSET parameter of EYU9XDUT 258





---

## Notices

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

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

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

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

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

**The following paragraph does not apply in the United Kingdom or any other country where such provisions are inconsistent with local law:**

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore this statement may not apply to you.

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

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

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

---

## Trademarks

IBM, the IBM logo, and [ibm.com](http://ibm.com) are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. A current list of IBM trademarks is available on the Web at Copyright and trademark information at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

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

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

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

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

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

Other product and service names might be trademarks of IBM or other companies.



---

# Readers' Comments — We'd Like to Hear from You

**CICS Transaction Server for z/OS  
Installation Guide  
Version 3 Release 2**

**Publication No. GC34-6812-04**

We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.

For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state on this form.

Comments:

Thank you for your support.

Submit your comments using one of these channels:

- Send your comments to the address on the reverse side of this form.
- Send a fax to the following number: +44-1962-816151
- Send your comments via email to: [idrcf@hursley.ibm.com](mailto:idrcf@hursley.ibm.com)

If you would like a response from IBM, please fill in the following information:

\_\_\_\_\_

Name

\_\_\_\_\_

Address

\_\_\_\_\_

Company or Organization

\_\_\_\_\_

Phone No.

\_\_\_\_\_

Email address



Fold and Tape

**Please do not staple**

Fold and Tape

PLACE  
POSTAGE  
STAMP  
HERE

IBM United Kingdom Limited  
User Technologies Department (MP095)  
Hursley Park  
Winchester  
Hampshire  
SO21 2JN  
United Kingdom

Fold and Tape

**Please do not staple**

Fold and Tape





Product Number: 5655-M15

GC34-6812-04





Spine information:



CICS Transaction Server for z/OS    Installation Guide

Version 3  
Release 2