

IMS/ESA



Installation Volume 2: System Definition and Tailoring

Version 6

IMS/ESA



Installation Volume 2: System Definition and Tailoring

Version 6

Note

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

Sixth Edition (July 2000) (Softcopy Only)

This edition replaces and makes obsolete the previous edition, GC26-8737-04. This edition is available in softcopy format only. The technical changes for this edition are summarized under "Summary of Changes" on page xix and are indicated by a vertical bar to the left of a change.

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

A form for readers' comments is provided at the back of this publication. If the form has been removed, address your comments to:

IBM Corporation, BWE/H3
P.O. Box 49023
San Jose, CA, 95161-9023
U.S.A.

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

© **Copyright International Business Machines Corporation 1974, 2000. All rights reserved.**

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

Contents

Notices	xi
Trademarks	xi
Product Names	xii
Preface	xiii
Summary of Contents	xiv
DB2 and CICS Information	xiv
Terminology and Related Publications	xiv
Change Indicators	xv
Syntax Diagrams	xv
Summary of Changes	xix
Changes to the Current Edition of the Book	xix
Changes to This Book for V6	xix
Library Changes for Version 6	xix
Chapter 1. Designing the IMS System	1
Macros and Procedures	1
Assigning System Resource Options with System Configuration Macros	2
Choosing the Number of Regions	3
Setting a Checkpoint Frequency	3
Selecting an IMS Lock Manager	4
Selecting the DL/I Separate Address Space	5
Security Options	7
Allocating MFS Pool Space	8
Defining Online Applications with Database and Application Macros	9
Declaring Online Databases	9
Declaring Online Programs	10
Defining IMS Transactions	12
Defining Terminals with Data Communication Macros	14
Defining Non-VTAM Terminals	14
Defining Switched Communication Devices	18
Specifying Macros for Multiple Systems Coupling	18
Defining VTAM Terminals	19
Specifying the Master Terminal	21
Including ETO in the IMS System	23
Including FDBR in the IMS System	25
DB/DC	25
DBCTL	25
Including RSR in the IMS System	25
Chapter 2. The System Definition Process	29
Selecting the Appropriate IMS System Definition	29
Using Large System Generation	30
Using DBCTL	30
Using DCCTL	30
Using the Macro Table	31
Managing System Definition	40
Sequencing Stage 1 Input	40
LGEN System Definition Changes	40
LGEN Stage 1 Processing	40
LGEN Stage 2 Processing	41
Verifying System Definition	41

Preprocessor	42
Executing the Preprocessor	43
Estimating Storage Requirements for the Preprocessor	45
Running System Definition Stage 1	47
Entering Stage 1 Input Statements	47
Stage 1 and Extended Terminal Option	48
Stage 1 Output	48
Running System Definition Stage 2	49
Stage 2 Output	49
Maintaining Stage 1 Output with SMP/E.	50
Large System Definition SMP/E JCLIN	50
Sample JCLIN Job	50
Chapter 3. Macros	53
Guidelines for IMS Macros.	54
Resource Naming Rules	54
Coding Conventions	56
APPLCTN Macro	56
Positional Parameters	57
Keyword Parameters.	58
BUFPOOLS Macro	61
Keyword Parameters.	62
COMM Macro	64
Keyword Parameters.	65
CONFIG Macro.	73
Label Field	73
Keyword Parameters.	73
CTLUNIT Macro	74
Keyword Parameters.	74
DATABASE Macro	75
Positional Parameter.	75
Keyword Parameters.	75
FPCTRL Macro.	76
Keyword Parameters.	77
IDLIST Macro	78
Label	79
Positional Parameters	79
IMSCTF Macro	79
Keyword Parameters.	80
IMSCTRL Macro	82
Keyword Parameters.	84
Guidelines for Selecting the System Definition Type	94
IMSGEN Macro.	95
Keyword Parameters.	99
Assembler and Linkage Editor Options	99
IMS Data Set Options	102
Job Control Language Statements	104
General Communication Options	106
Security Options	107
Sample IMSGEN Macro Statement	107
LINE Macro	108
Keyword Parameters	109
LINEGRP Macro	112
Keyword Parameters	113
MSGQUEUE Macro.	116
Keyword Parameters	116

MSLINK Macro	120
Keyword Parameters	121
MSNAME Macro	122
Label Field	122
Keyword Parameters	123
MSPLINK Macro	123
Label Field	124
Keyword Parameters	124
NAME Macro	127
Positional Parameters	129
Keyword Parameters	131
POOL Macro	132
RTCODE Macro	132
Keyword Parameters	132
SECURITY Macro	133
Keyword Parameters	134
Relationships between the SECURITY, COMM, and IMSGEN Macros	137
SPAREA Macro	137
STATION Macro	137
Keyword Parameters	138
SUBPOOL Macro	139
Keyword Parameters	140
TERMINAL Macro	140
Label Field	153
Keyword Parameters	153
TRANSACT Macro	173
Keyword Parameters	176
TYPE Macro	186
Keyword Parameters	187
VTAMPOOL Macro	188
Chapter 4. Procedures	189
Using IMS Sysgen-Supplied Procedures	191
Storing Procedures	191
Allocating Terminal Devices in an Online IMS System	192
Specifying the DFSRESLB DD Statement in IMS Batch Procedures	192
DBBBATCH Procedure	192
Procedure Description	193
DBBBATCH Parameters	194
DBBBATCH DD Statements	194
DBC Procedure	195
Procedure Description	195
DBC Parameters	199
DBC DD Statements	199
DBRC Procedure	200
Procedure Description	200
DBRC Parameters	201
DBRC DD Statements	201
DCC Procedure	201
Procedure Description	201
DCC Parameters	209
DCC DD Statements	210
DFSMPR Procedure	211
Procedure Description	211
DFSMPR Parameters	211
DFSMPR DD Statements	212

DLIBATCH Procedure	212
Procedure Description	212
DLIBATCH Parameters	213
DLIBATCH DD Statements	213
DLISAS Procedure	214
Procedure Description	214
DLISAS Parameters	215
DLISAS DD Statements	216
DXRJPROC Procedure	216
Procedure Description	216
Startup Procedure for IRLM 2.1	216
DXRJPROC Parameter Descriptions for IRLM 2.1	217
DXRJPROC DD Statements for IRLM 2.1	217
FDR Procedure	217
Procedure Description	217
FDR Procedure Parameters	220
FDR Procedure DD Statements	220
FPUTIL Procedure	221
Procedure Description	221
FPUTIL Procedure Parameters	221
IMS Procedure	221
Procedure Description	221
IMS Procedure Parameters	230
IMS Procedure DD Statements	231
IMSBATCH Procedure	232
IMSBATCH Procedure Parameters	232
IMSBATCH Procedure DD Statements	233
IMSCOBGO Procedure	233
Procedure Description	233
IMSCOBGO Procedure Parameters	235
IMSCOBGO Procedure DD Statements	235
IMSCOBOL Procedure	235
Procedure Description	235
IMSCOBOL Procedure Parameters	236
IMSCOBOL Procedure DD Statements	236
IMSFP Procedure	236
Procedure Description	236
IMSFP Procedure Parameters	237
IMSFP Procedure DD Statements	237
IMSMMSG Job	237
Job Description	237
IMSPLI Procedure	238
IMSPLI Procedure Parameters	238
IMSPLI Procedure DD Statements	238
IMSPLIGO Procedure	239
Procedure Description	239
IMSPLIGO Procedure Parameter Descriptions	240
IMSPLIGO Procedure DD Statements	241
IMSRDR Procedure	241
Procedure Description	241
IMSRDR Procedure Parameter Descriptions	241
IMSRDR Procedure DD Statements	241
RDIBATCH Procedure	242
Procedure Description	242
RDIBATCH Procedure Parameter Descriptions	243
RDIBATCH Procedure DD Statements	243

DL/I Interfaces	243
CBLTDLI	243
PLITDLI	243
Parameter Descriptions	244
DD Statement Descriptions	279
Chapter 5. IMS Dumping and Dump Formatting Options	287
Installing the Interactive Dump Formatter	287
Dumping Options	287
MVS Dump Options	288
Online System Dumps.	288
Online System Dependent Region Dumps	291
Batch System Dumps	291
IRLM Use of SDUMP	293
Chapter 6. Installing the Transport Manager Subsystem	295
VTAM Definition for the Transport Manager Subsystem	295
Chapter 7. Tailoring the IMS System to Your Environment	299
Tailoring the IMS Procedure Library	299
Procedures Generated by System Definition	299
Altering the IMS Procedure Library	300
Specifying IMS PROCLIB Members	304
Member DBFMSSDBx	304
Member DFSDCxxx	305
Member DFSDRFnn	306
Member DFSFDRxx	307
Member DFSFIXnn	309
Member DFSHSBxx	313
Member DFSINTxx	318
Member DFSMPLxx	318
Member DFSPBDBC	320
Member DFSPBDCC	320
Member DFSPBIMS	321
Member DFRSRxx	321
Member DFSSPMxx	332
Member DFSSQxxx	336
Member DFSVFLqq	338
Member DFSVSMxx	339
Member DFSYDTx	363
Member DFS62DTx	363
Specifying IMS Execution Parameters	365
Database and PSB Exec Parameters for the Control Region	365
Data Communication EXEC Parameters for the Control Region	367
System Control and Performance EXEC Parameters for the Control Region	367
Recovery-Related EXEC Parameters for the Control Region.	370
Security-Related EXEC Parameters for the Control Region	370
IMS Message Processing Region Parameters	371
IMS Batch Message Processing Region Parameters	375
Specifying IMS Buffer Pools	378
VSAM Subpool Definition	379
OSAM Subpool Definition	379
OSAM Buffer Pool Compatibility Definition	380
Specifying VSAM and OSAM Subpools	381
Specifications for OSAM Sequential Buffering	381
Making IMS and IMSRDR Procedures Accessible to MVS	382

Organizing PL/I Modules for Use with the PL/I Optimizer	382
Coding COBOL Programs That Run under IMS	382
Specifying Sequential Buffering Control Statements	383
Syntax of SB Control Statements.	383
SB Parameters (SBPARAM) Control Statement	384
SB Image Capture (SBIC) Control Statement	387
SB Compare Option (SBCO) Control Statement	387
SB Snap (SBSNAP) Control Statement	388
SB Evaluation Snap (SBESNAP) Control Statement.	389
Snap Destination (SNAPDEST) Control Statement	390
Specifying High-Speed Sequential Processing Control Statements	391
Syntax of HSSP Control Statements	391
Set Options (SETO) Control Statement	391
Set Range (SETR) Control Statement	393
HSSP Control Statement Examples	394
Supporting CCTL Users with DBCTL Databases	395
Preparing a CCTL	396
The DRA Startup Table	396
Enabling ETO Support for ACF/VTAM Terminals	399
ETO Descriptors	399
Logon Descriptors	401
MFS Device Descriptors	405
MSC Descriptors.	406
User Descriptors	406
Specifying the Number of Hash Table Slots	409
Chapter 8. Accessing IMS Databases via the ODBA Interface	411
Overview.	411
Setting Up the DRA and the ODBA Interface	411
Creating the ODBA DRA Start-up Table	411
Loading and Running the ODBA and DRA Modules in the OS/390	
Application Region	412
Linking Application Programs	412
Establishing and Defining Security	412
Chapter 9. Accessing IMS Databases with CICS	415
Installing Data Sharing with CICS	415
Installing CICS Intersystem Communication or Multiregion Operation with DL/I	
Buffer Handler Pool	415
Installing CICS for Use with IMS Intersystem Communication	416
Coding Considerations for PSBs	416
Using Sequential Buffering	417
CICS Connected to DL/I	417
Accessing DBCTL Databases with CICS as a CCTL.	417
DBCTL Tasks	417
CICS Tasks.	418
Chapter 10. Accessing DB2 Databases with IMS	419
What You Should Do for Your IMS Subsystem	419
Defining Your DB2 Subsystems to IMS	419
Defining the Language Interface Module	423
Specifying the Desired Trace Options	424
Specifying DB2 for IMS Batch Regions	424
What You Should Do for Your DB2 Subsystem.	425
Providing Appropriate Tables in DB2	425
Placing DB2 Modules and Tables in the Appropriate Library	425

Appendix A. IMS System Definition (Step 3) Examples	427
IMS DB/DC Environment	427
Data Communication Characteristics	427
System Configuration Macro Statements	428
Database and Application Macro Statements	429
Data Communication Macro Statements	434
BTAM, BSAM, GAM, and ARAM Macro Sets	434
Switched Communication Device Macro Sets	440
VTAM Macro Sets	442
System Configuration Macro Statement	449
IMS Multiple Systems Coupling	450
Data-Sharing System Configuration	455
Example 1. Data Sharing at the Database-Level with Update Access	455
Example 2. Data Sharing at the Database Level with Read Access	456
Example 3. Intra-CPC Block-Level Data Sharing	457
Example 4. Inter-CPC Block-Level Data Sharing	458
IMS DBCTL Environment	458
Appendix B. MFS System Definition and Programming Considerations	461
System Definition Macro Instructions	461
BUFPOOLS Macro	461
MSGQUEUE Macro	461
IMSGEN Macro	461
COMM Macro	461
TYPE, TERMINAL, and CONFIG Macros	462
System Programming Considerations	464
INDEX Directory	464
FORMAT, FORMATA, FORMATB, and TFORMAT Library Directories	464
Use of Concatenated Format Libraries	464
Use of the MFS Buffer Pool	465
Bibliography	467
IMS/ESA Version 6 Library	467
Index	469

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. 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. Subject to IBM's valid intellectual property or other legally protectable rights, any functionally equivalent product, program, or service may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter 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
500 Columbus Avenue
Thornwood, NY 10594
U.S.A.

Licensees of this program who wish to have information about it for the purpose of enabling (1) the exchange of information between independently created programs and other programs (including this one) and (2) the mutual use of the information that has been exchanged, should contact:

IBM Corporation, Department W92/H3
P.O. Box 49023
San Jose, CA 95161-9023

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

ACF/VTAM	IMS
CICS	IMS/ESA
CICS/ESA	MVS
CICS/MVS	MVS/ESA
DATABASE 2	OS/390
DB2	RACF
Hiperspace	Resource Measurement Facility
IBM	RMF
	VTAM

The term "Touch-Tone," is a trademark of American Telephone & Telegraph Company. Other company, product, and service names, which may be denoted by a double asterisk (**), may be trademarks or service marks of others.

Product Names

In this book, the following licensed programs have shortened names:

- “DB2 for OS/390” is referred to as “DB2.”.
- “COBOL for MVS & VS” is referred to as “COBOL”.
- “PL/I for MVS & VS” is referred to as “PL/I”.

Preface

This book is one of two books that help you install an IMS/ESA system. The other book is *IMS/ESA Installation Volume 1: Installation and Verification*.

This book provides detailed reference information on macros, procedures, and other system-oriented information to help you install and make modifications to IMS/ESA. (IMS/ESA is hereafter referred to as IMS.) The following table lists the subtasks associated with the IMS installation task and identifies the location of key information about these subtasks.

Table 1. *IMS Installation Subtasks.* In this table, 'Volume 1' refers to *IMS/ESA Installation Volume 1: Installation and Verification* and 'Volume 2' refers to *IMS/ESA Installation Volume 2: System Definition and Tailoring*

Installation Subtask	Location of Information
Installing an IMS system	<i>Volume 1</i>
Verifying the correct installation of an IMS system	<i>Volume 1</i>
Changing your IMS system—after it has been installed and verified— as follows: Installing Internal Resource Lock Manager (IRLM) Installing Extended Terminal Option (ETO) feature Installing service, using pregeneration mode Deleting device support	<i>Volume 1</i>
Using the IVP system to test application or service changes	<i>Volume 1</i>
Using the IVP system for demonstrations , in-house training , or developing operation and recovery procedures	<i>Volume 1</i>
Building or moving your own systems onto a copy of the IVP system	<i>Volume 1</i>
Using the DL/I Test Program and IVP to experiment with different DL/I call sequences	<i>Volume 1</i>
Assigning system resource options with system configuration macros	<i>Volume 2</i>
Defining online applications with database and application macros	<i>Volume 2</i>
Defining terminals with Data Communication macros	<i>Volume 2</i>
Including ETO or Remote Site Recovery (RSR) in the IMS system	<i>Volume 2</i>
Implementing the system definition process	<i>Volume 2</i>
Installing the Transport Manager subsystem	<i>Volume 2</i>
Tailoring the IMS system for your environment	<i>Volume 2</i>
Accessing the IMS databases with CICS or DB2 database with IMS	<i>Volume 2</i>

As you look at this table of installation-related activities, notice that *IMS/ESA Installation Volume 1: Installation and Verification* is primarily used during the period when you are actually installing a new system and ensuring that it has been installed properly. To tune and tailor on an ongoing basis throughout the life of this IMS system, you rely primarily on information in *IMS/ESA Installation Volume 2: System Definition and Tailoring*.

Therefore, if you are responsible for installing a new IMS system, you should have copies of both volumes of *IMS/ESA Installation*. If you are responsible for supporting an already-installed IMS system, you probably need access to *IMS/ESA Installation Volume 2: System Definition and Tailoring* only.

Summary of Contents

This book has nine chapters:

“Chapter 1. Designing the IMS System” on page 1, describes the macro statements that define your system. These macro statements include system configuration macros, database and application macros, and data communication macros. This chapter also describes how to use these macros to define online applications and IMS terminals, and how to assign system resource options.

“Chapter 2. The System Definition Process” on page 29, describes the major steps of the system definition process. These steps include selecting, managing, and verifying IMS system definition. This chapter also describes running system definition stage1 and stage2.

“Chapter 3. Macros” on page 53, provides the resource naming rules, coding conventions, and syntax for all macros used in IMS system definition.

“Chapter 4. Procedures” on page 189, identifies the cataloged procedures, control statements, and jobs that are added to PROCLIB during system definition.

“Chapter 5. IMS Dumping and Dump Formatting Options” on page 287, provides information on IMS dumping and dump formatting options.

“Chapter 6. Installing the Transport Manager Subsystem” on page 295, describes how to install the transport manager subsystem.

“Chapter 7. Tailoring the IMS System to Your Environment” on page 299 shows what you can do to improve your system’s performance. The chapter also includes a database control (DBCTL) section for coordinator controller subsystem (CCTL) users.

“Chapter 8. Accessing IMS Databases via the ODBA Interface” on page 411, explains how to use the Open Database Access (ODBA) callable interface that enables any OS/390 application running in a recoverable, resource-managed OS/390 address space to issue DL/I database calls to an IMS DB subsystem.

“Chapter 9. Accessing IMS Databases with CICS” on page 415, explains data sharing with CICS and intersystem communications with CICS. This chapter includes a brief description of CICS via DBCTL.

“Chapter 10. Accessing DB2 Databases with IMS” on page 419, describes installation tasks required for IMS support of DB2 databases.

DB2 and CICS Information

Only the DB2 and CICS information that applies to IMS installation is included in this manual.

For CICS customers, whenever CICS is mentioned, this book refers to CICS via the Database Resource Adapter (DRA).

Terminology and Related Publications

For a list of related publications, refer to “Bibliography” on page 467.

Related Reading: For definitions of terminology used in this manual and references to related information in other IMS manuals, see *IMS/ESA Master Index and Glossary*.

Change Indicators

Technical changes are indicated in this publication by a vertical bar (|) to the left of the changed text. If a figure has changed, a vertical bar appears to the left of the figure caption.

Syntax Diagrams

The following rules apply to the syntax diagrams used in this book:

Arrow symbols

Read the syntax diagrams from left to right, from top to bottom, following the path of the line.

- ▶▶— Indicates the beginning of a statement.
- ▶ Indicates that the statement syntax is continued on the next line.
- ▶— Indicates that a statement is continued from the previous line.
- ▶◀ Indicates the end of a statement.

Diagrams of syntactical units other than complete statements start with the ▶— symbol and end with the —▶ symbol.

Conventions

- Keywords, their allowable synonyms, and reserved parameters, appear in uppercase for MVS and OS/2 operating systems, and lowercase for UNIX operating systems. These items must be entered exactly as shown.
- Variables appear in lowercase italics (for example, *column-name*). They represent user-defined parameters or suboptions.
- When entering commands, separate parameters and keywords by at least one blank if there is no intervening punctuation.
- Enter punctuation marks (slashes, commas, periods, parentheses, quotation marks, equal signs) and numbers exactly as given.
- Footnotes are shown by a number in parentheses, for example, (1).
- A `␣` symbol indicates one blank position.

Required items

Required items appear on the horizontal line (the main path).

▶▶—REQUIRED_ITEM————▶▶

Optional Items

Optional items appear below the main path.

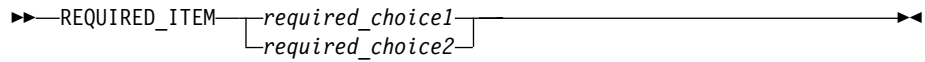
▶▶—REQUIRED_ITEM—optional_item————▶▶

If an optional item appears above the main path, that item has no effect on the execution of the statement and is used only for readability.

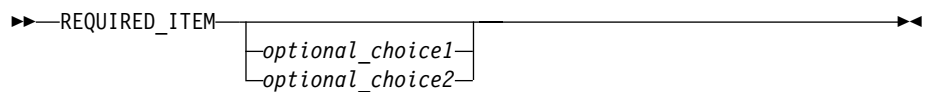


Multiple required or optional items

If you can choose from two or more items, they appear vertically in a stack. If you *must* choose one of the items, one item of the stack appears on the main path.

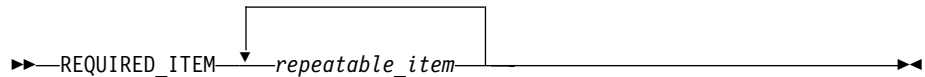


If choosing one of the items is optional, the entire stack appears below the main path.

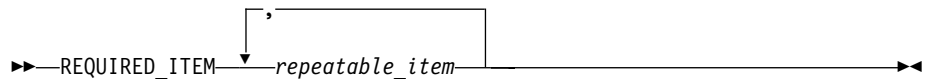


Repeatable items

An arrow returning to the left above the main line indicates that an item can be repeated.



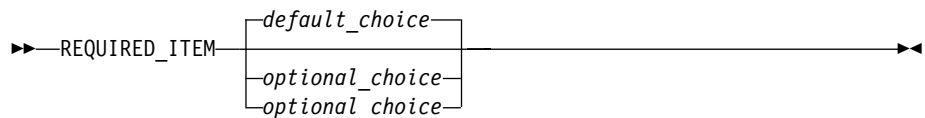
If the repeat arrow contains a comma, you must separate repeated items with a comma.



A repeat arrow above a stack indicates that you can specify more than one of the choices in the stack.

Default keywords

IBM-supplied default keywords appear above the main path, and the remaining choices are shown below the main path. In the parameter list following the syntax diagram, the default choices are underlined.



IMS-specific syntax information

Fragments

Sometimes a diagram must be split into fragments. The fragments are represented by a letter or fragment name, set off like this: | A |. The fragment follows the end of the main diagram. The following example shows the use of a fragment.

▶▶—STATEMENT—*item 1*—*item 2*—| A |————▶▶

A:

————*item 3*—KEYWORD—	————
————*item 4*—	————
————*item 5*—	————
————*item 6*—	————

Substitution-block

Sometimes a set of several parameters is represented by a substitution-block such as <A>. For example, in the imaginary /VERB command you could enter /VERB LINE 1, /VERB EITHER LINE 1, or /VERB OR LINE 1.

▶▶—/VERB—|————|————LINE—*line#*————▶▶
|————<A>————|

where <A> is:

▶▶—|————|————▶▶
|————EITHER————|
|————OR————|

Parameter endings

Parameters with number values end with the symbol '#', parameters that are names end with 'name', and parameters that can be generic end with '*'.

▶▶—/MSVERIFY—|————MSNAME—*msname*————|
|————SYSID—*sysid#*————|

The MSNAME keyword in the example supports a name value and the SYSID keyword supports a number value.

Summary of Changes

Changes to the Current Edition of the Book

This edition, which is softcopy only, includes technical and editorial changes.

Changes to This Book for V6

The organization of this book has changed from Version 5 in these respects:

- “Chapter 1. Designing the IMS System” on page 1 now combines information from Version 5’s Chapter 1 “IMS/ESA System Definition” and Chapter 2 “Designing the IMS System”.
- “Chapter 4. Procedures” on page 189 has been substantially reorganized:
 - Parameter descriptions for all procedures are now compiled in a single alphabetic list under the topic “Parameter Descriptions” on page 244.
 - DD statement descriptions for all procedures are now compiled in a single alphabetic list under the topic “DD Statement Descriptions” on page 279.
 - Information on “IMS Dumping and Dump Formatting Options” is now a separate chapter, Chapter 5.

This book contains new information on the following topics:

- Block-level sharing of VSO DEDB areas
- Fast database recovery
- Shared queues in a sysplex
- OSAM database coupling facility caching
- Fast database recovery in a sysplex
- Open Database Access (ODBA)

Library Changes for Version 6

The IMS/ESA Version 6 library differs from the IMS/ESA Version 5 library in these major respects:

- *IMS/ESA Common Queue Server Guide and Reference*
This new book describes the IMS Common Queue Server (CQS).
- *IMS/ESA DBRC Guide and Reference*
This new book describes all the functions of IMS Database Recovery Control (DBRC).
- The IMS Application Programming summary books (*IMS/ESA Application Programming: Database Manager Summary*, *IMS/ESA Application Programming: Transaction Manager Summary*, and *IMS/ESA Application Programming: EXEC DLI Commands for CICS and IMS Summary*) are no longer included with the IMS library.
- The Softcopy Master Index is not included.
- All information about IRLM 1.5 and data sharing using IRLM 1.5 has been removed from the IMS V6 books. If you use IRLM 1.5, and want to migrate to using IRLM 2.1 and Sysplex data sharing, see *IMS/ESA Release Planning Guide*.
- The chapter that was titled “Database Control (DBCTL) Interface” in the *IMS/ESA Customization Guide* has been revised for Open Database Access (ODBA) and moved to “Appendix A, Using the Database Resource Adapter (DRA)” in the *IMS/ESA Application Programming: Database Manager*.

Chapter 1. Designing the IMS System

Before you can use the Information Management System/Enterprise Systems Architecture Transaction Manager (IMS/ESA TM) and the Information Management System/Enterprise Systems Architecture Database Manager (IMS/ESA DB), you must define the elements and functions that make up an IMS system. These include:

- Databases
- Application programs
- Terminals

IMS provides macros and procedures that enable you to define your system. The IMS macros, as well as coding and usage rules, are described in “Chapter 3. Macros” on page 53. Procedures are described in “Chapter 4. Procedures” on page 189.

In this Chapter:

- “Macros and Procedures”
- “Assigning System Resource Options with System Configuration Macros” on page 2
- “Defining Online Applications with Database and Application Macros” on page 9
- “Defining Terminals with Data Communication Macros” on page 14
- “Including ETO in the IMS System” on page 23
- “Including FDBR in the IMS System” on page 25
- “Including RSR in the IMS System” on page 25

Macros and Procedures

IMS provides a set of macro statements for system definition. You use macro statements to generate a sequence of assembler statements from a single source statement such as APPLCTN or TERMINAL. For example, to describe program resource requirements for your application programs, you use the APPLCTN macro.

IMS uses three types of macro statements to define your system:

- System configuration
- Database and application
- Data communication

Table 2 on page 2 is an introductory list of IMS macros; for detailed information on system-specific information, see Table 12 on page 32. Examples of macro statement sets are shown in “Appendix A. IMS System Definition (Step 3) Examples” on page 427.

Designing the System

Table 2. Use of System Definition Macros

Type of Macro Statement		Macro Name	Resources Each Macro Defines
System Configuration Macros		BUFPOOLS FPCTRL IMSCTF IMSCTRL IMSGEN MSGQUEUE ¹ SECURITY	Buffer pool sizes Fast Path options System control integrity factors Library naming, JCL options, and ETO feature MVS system and control options Message queue device and message sizes Security options
Database and Application Macros		APPLCTN DATABASE RTCODE ¹ TRANSACTION ¹	Program characteristics and PSB names DBD names Fast Path message routing Transaction processing options
Data Communication Macros ¹	BTAM, BSAM, GAM, ARAM	COMM CTLUNIT LINE LINEGRP NAME STATION TERMINAL	General communication options 2848, 2972, 3271 control Communication line to IMS DD names for polled devices LTERM definition System/3 and System/7 line groups Physical characteristics of terminal
	Switched Communication Device	CONFIG IDLIST LINE LINEGRP NAME POOL SUBPOOL TERMINAL	Switched 3275 configuration Security for switched 3275 Communication line to IMS DD names for polled devices LTERM definition Set of LTERMs for switched line group Set of LTERMs for a switched line terminal Physical characteristics of terminal
	Multiple System Coupling	MSLINK MSNAME MSPLINK NAME	Logical Link names Partner systems Physical link names LTERM definition
	VTAM	COMM NAME SUBPOOL TERMINAL TYPE VTAMPOOL	General communication options LTERM definition Set of LTERMs for a static ISC session Physical characteristics of a static terminal VTAM terminal type Multiple systems with ISC sessions

You can tailor the IMS system to suit your installation's requirements with members of the IMS procedure library (IMS.PROCLIB). With these procedures, you can, for example:

- Initialize a DBCTL environment (DBC)
- Initialize a DB/DC environment (IMS)
- Initialize a DCCTL environment (DCC)
- Execute a Fast Path application program (IMSFP)

Assigning System Resource Options with System Configuration Macros

The system configuration macro statements describe the basic IMS control program options. The macros are as follows:

¹. Not valid in a DBCTL environment.

Assigning System Resource Options

Macro	Purpose
BUFPOOLS	Defines the storage buffer pool sizes for the online system control program region.
FPCTRL	Defines the Fast Path options of the IMS control program.
IMSCTF	Defines additional options and system parameters under which IMS is to operate.
IMSCTRL	Defines the basic IMS control program options under which IMS is to operate. The ETO feature can optionally be included.
IMSGEN	Defines the assembler and linkage editor data sets and options, as well as the required system definition and output options.
MSGQUEUE	Defines the characteristics of the three user message queue data sets.
SECURITY	Defines the security options to be in effect during IMS execution unless they are overridden when IMS is started.

A group of parameters within the system configuration macro set relates to the integrity of the IMS online system. These parameters specify resources available for checkpoints and program isolation. The allocation of system data sets and control program storage depends upon the expected processing load and the strategy for recovery of the online IMS system.

Choosing the Number of Regions

Overall throughput is conditionally based on the number of active dependent regions and the availability of shared resources for database processing. A processing program must have an available message region before it can enter the scheduling selection logic.

Use the MAXREGN keyword on the IMSCTRL macro to specify a working set of BMP and message regions. Up to a maximum of 999 other dependent regions can be dynamically allocated by the MTO using the /START command. Your operations procedure stipulates how many regions are active and when they are active. The scheduling algorithm controls the actual programs that execute in the regions in response to enqueued messages.

For DBCTL, the maximum sum of BMPs and DRA threads is 999.

Setting a Checkpoint Frequency

IMS records information for restarting interrupted operations by using checkpoints. When an operation is interrupted, reprocessing occurs from the point of system interruption to a forward continuation point. Using the status information captured during checkpoint, IMS can restore the content of the message queues and database changes.

2. If you are installing an IMS batch system, the IMSCTRL and IMSGEN macros are the only two that are required. IMSCTF is optional, but it contains some parameters that can be useful in your database system, and you should consider specifying it. SECURITY is also optional.

Assigning System Resource Options

Recommendation: It is strongly recommended that you take frequent checkpoints in order to minimize the amount of time it takes to do the restart. The trade-off for taking frequent checkpoints (and efficient restart) is an increase in processing overhead.

Begin a restart at the current checkpoint minus one. This applies to restarts from all checkpoints except RESTART, SNAPQ, and SHUTDOWN.

Use the CPLOG keyword on the IMSCTF macro to control the frequency of IMS internal checkpoints. The value you choose for this keyword represents the number of additional system log records that are created before a new checkpoint is invoked. When the increment exceeds the CPLOG value, checkpoint processing begins. The default is 1000. IMS system checkpoints are invoked by the /CHECKPOINT command or by application programs that have been authorized to enter the /CHECKPOINT command.

One technique for arriving at a CPLOG value is to use the transaction profiles and estimate, for a small peak processing interval, the number of system log events. The number of log records exceeds the total of the following:

- (Number of input transactions) x 2
- (Number of output messages) x 3
- (Number of secondary transactions) x 2
- (Number of database update calls for REPL,DLET)
- (Number of database inserts) x 2
- (Number of related index pointer maintenance calls) x 2
- (Number of programs scheduled and terminated)
- (Instances of dial lines connected and disconnected)
- (Number of checkpoint calls from programs) x 3

To insure an accurate estimate, your interval should be large enough to extend beyond the estimated response time for at least one of the longest-running transactions.

You can adjust the checkpoint frequency after observing the stability of the online IMS system. Use IMS Monitor reports to assess the processing overhead. The Region Summary report shows the total elapsed time and average elapsed time taken for checkpoints in the trace interval.

Selecting an IMS Lock Manager

To protect database integrity, an IMS system serializes requests for database resources to ensure that no two application programs are allowed to update a database segment concurrently. Lock management is the process of controlling concurrent requests. To control the stored information about the requests, you can use either or both of the following methods:

- A program isolation lock manager
- The services of the Internal Resource Lock Manager (IRLM) component

The former only controls lock requests for a single IMS system (local locking); the latter controls requests for multiple IMS systems (global locking), as well as for a single system.

Monitoring program isolation activity is an integral part of the IMS Monitor, which is invoked by the /TRACE command. A separate program isolation trace can also be

Assigning System Resource Options

used to analyze performance. The advantage of selecting program isolation to manage locks is in not having to plan any special operating and recovery procedures, as is necessary for the IRLM.

To indicate that an IRLM is to manage locking, specify the IRLM's MVS subsystem name on the IRLMNM parameter of the IMSCTRL macro. The IRLM is required if the IMS online system is to take part in block-level data sharing. Because the IRLM executes in a separate region, you can plan for a known amount of common storage area (ECSA) to be allocated for use by the IRLM component, or you can retain most of its control blocks in local storage rather than in the common storage area.

Your choice of lock manager is not necessarily permanent. Using parameters on the EXEC statement for the control region, you can override the use of program isolation locking. Specify IRLM=Y and give the IRLM MVS subsystem name on the IRLMNM parameter to indicate this change. In this way, you can allow for the IMS online system to take part in block-level data sharing.

Related Reading: For information on administering a data-sharing environment, see *IMS/ESA Administration Guide: System* for details on how to plan for and initialize operating procedures for an IRLM.

Selecting the DL/I Separate Address Space

You can use a separate address space to contain DL/I code, control blocks, and buffers for full-function databases. To do this, use the local storage option (LSO). Specify LSO=S as an EXEC parameter for the control region to indicate the choice of a DL/I address space.

For the DBCTL environment, a separate DL/I address space (DLISAS) is required, and the LSO parameter is not supported.

In a DB/DC environment, if a DBCTL function is used, you must specify LSO=S.

DLISAS must also be selected for an RSR tracking subsystem that does Database Level Tracking (DLT) for full function databases.

Storage Considerations

With the LSO=S option, storage is moved to a separate address space. Additionally, most of the PSB pool and the DMB pool are moved to the DL/I address space. These pools are not moved as a result of the LSO=Y option.

The IMS control program automatically initiates the DL/I address space. If either the control or DL/I address space terminates, the other is automatically terminated.

MVS cross-memory services process an application database call. Because of the frequency with which cross-memory operations are used, appropriate hardware support is recommended.

IMS restart procedures are insensitive to the LSO specification. For example, an LSO=S system can be terminated, and IMS restart procedures can be performed on a system specifying LSO=Y.

The LSO=S option moves the following major storage into the DL/I separate address space:

- DL/I code

Assigning System Resource Options

- Database buffers
- DMB pool, both resident and nonresident
- DMB work pool
- Most of the PSB pool, both resident and nonresident
- For non-Fast Path systems:
 - PI ENQ/DEQ tables

For an LSO=S system, the following storage is in the MVS common area:

- OSAM code
- Log buffers
- Resident intent lists

The resident intent lists are in the control address space for all configurations other than LSO=S.

PSB Pool Definition

In an LSO=S system, two PSB pools exist: one in the MVS common area (DLMP) and one in the DL/I address space (DPSB). The DPSB pool contains the DL/I control blocks associated with a full-function database PCB (the JCB for example). The DLMP pool contains the communications PCBs, Fast Path PCBs, and the key feedback area of the full-function PCBs.

Specify the sizes of these pools with the SASPSB parameter on the BUFPOOLS macro. You can override these sizes with the CSAPSB and DLIPSB parameters on the IMS procedure. The command /DISPLAY P00L PSBP displays usage information for both PSB pools in an LSO=S system.

The output of the ACBGEN utility indicates, for a given PSB, the amount of space required in each pool. An out-of-space condition in the DPSB pool or a wait-for-storage condition is reflected in output reports of the IMS Monitor. In these reports, 'DLMP' and 'DPSB' represent the CSA PSB pool and the DLS PSB pool, respectively.

The ACBGEN utility organizes the PSB so that all data for the DLMP pool precedes all data for the DPSB pool. This organization applies to all IMS configurations.

If LSO=S is not being used, a single PSB pool is in the MVS common area. In this case, the PSB parameter on the BUFPOOLS macro and on the IMS start procedure specifies the size of this pool.

To page fix the PSB pool, specify POOLS=DLMP and POOLS=DPSB in the DFSFIXxx member in IMS.PROCLIB. To fix the resident PSB pool, include the following module names in this member: DFSPSBRs and DFSDLIRS.

Invoking the DL/I Address Space

IMS system generation produces the skeletal procedure DLISAS and places it in IMS.PROCLIB. You should modify this procedure as necessary and copy it to SYS1.PROCLIB.

Use the DLINM parameter on the IMSCTRL macro and the IMS start procedure to rename the procedure.

DLISAS Procedure Modifications: You need to plan for the following modifications to the DL/I address space procedure:

Assigning System Resource Options

- The JCL DD statements for the full-function databases must be in the DL/I address space procedure and not in the IMS procedure. JCL DD statements for Fast Path databases and the IMS system data sets remain in the IMS procedure.
- No changes to the dynamic allocation parameter lists in IMS.RESLIB are required.
- The specification of the active and inactive ACBLIBs must be identical in both the IMS procedure and the DLISAS procedure. (Both the control region and the DL/I address space read ACBLIB.) The data sets used must have disposition SHR, and the concatenation order must be identical.
- IMS.PROCLIB must be defined in both the IMS and DLISAS procedures.
- The only parameters passed on the DLISAS procedure are the region type (DLS) and, optionally, the IMSID of the control program to be connected. Pool sizes and database buffering options are specified on the IMS start procedure.

DL/I Exit Modifications: In an LSO=S system, the DL/I exits are entered in cross-memory mode.

Restriction: The following operating system restrictions apply to this configuration:

- The DL/I exits are not able to address storage in the control address space.
- If the DL/I exits use the IMS ISWITCH service, they must be changed to perform correctly in an LSO=S system. A macro specification of TO=DLI, not TO=DLI, must be specified on ISWITCH. The TO=DLI specification performs correctly in all IMS configurations.

Other Planning Considerations

Accounting Procedures with an LSO=Y System: Accounting procedures based on the IMS system log, Resource Measurement Facility (RMF), or System Management Facility (SMF) are affected when LSO= is specified. This data reflects the existence of multiple address spaces.

The CPUTIME value recorded in the IMS system log is affected by the LSO option specified. If LSO=Y is specified, only application program processing time is recorded in the type07 log record; most DL/I processing time is not included. For some applications DL/I processing time can represent a large percentage of the total processing time. If LSO=S is specified, DL/I processing time is included in CPUTIME. Because the IMS system log does not indicate which LSO option was specified, CPUTIME values should not be used for accounting or comparison purposes when switching options.

For SMF, full-function databases are accounted for in the DL/I address space. Fast Path databases and the IMS system data sets are accounted for in the control region.

Security Considerations: If the full-function databases are RACF protected, the DLISAS procedure must be authorized to access these resources.

Tuning Considerations: You must consider that the IMS control program consists of more than one address space. You might wish to include the control region and the DL/I address space in the same RMF performance group so that RMF statistics can be compared with systems that do not use the LSO=S option.

Security Options

The IMS system provides several ways to prevent unauthorized use of a terminal or connected device, as well as to control what processing is allowed in a dependent

Assigning System Resource Options

region. You can authorize the use of resources by declaring the resource name (such as the LTERM name or transaction code) in the input to the Security Maintenance utility.

Different declarations can be made depending on the IMS security function selected. The type of security can be specified using the TYPE parameter of the SECURITY macro. For a description of the TYPE parameter, see the SECURITY macro. The parameter values to invoke security checking are SIGNEXIT, TRANEXIT, and AGNEXIT. If you plan to use the RACF licensed program for your MVS system, specify RACFTERM, RACFCOM, or RACFAGN. For several of the parameters, you can specify values that determine whether the MTO can override the security matrix status at restart time. These same keywords define the MTO's flexibility in invoking or overriding the type of security checking that is active during the current online execution cycle. Refer to Table 12 on page 32 for a description of the options available to you, depending on your particular system's configuration.

The specification of the security options needs to be a part of the overall security design.

Related Reading: Refer to *IMS/ESA Administration Guide: System* for a complete description of design and operational considerations for security with IMS.

Allocating MFS Pool Space

Several parameters of the BUFPOOLS macro allow you to reserve storage for efficient processing of communications activity. The amount of storage required primarily depends upon how much concurrent use of message format blocks is expected and how many lines are active.

Defining Pool Space for Message Format Service (MFS) Devices

You can control the size of the message format pool and the number of fetch request elements (FREs) with the BUFPOOLS macro. One FRE is required to control each active block. Without a FRE, space cannot be assigned from the message format pool. When estimating the number of FREs, you should allow at least ten additional FREs to handle system activity. This eliminates the possibility of delayed response to other terminals. If preallocated FREs are not available, a dynamic FRE is allocated from the general area of the pool. However, dynamic FREs tend to fragment the pool, and their use should be avoided whenever possible.

To determine adequate size for the message format pool, compile a list of the MFS blocks, arranging them in DIF/MID and DOF/MOD pairs. Then, record their sizes and some indication of their frequency (for example, whether they are used by priority or quick response transactions). Allow space for input/output pairs to be in the pool. Watch for large messages or those with multiple segments, because these might preempt small, frequent messages from finding space for their format blocks in the pool, and this would directly affect their response time.

If you use the Message Format Service utility to build an index of the DASD addresses that reside in the message format pool during online execution, the index entries also reside in extended private storage.

You can override the amount of storage for the format pool and the number of FREs using the FBP and FRE parameters in the EXEC statement for the control region.

Related Reading: For more information about the Message Format Service utility, see *IMS/ESA Utilities Reference: Transaction Manager*.

Defining Online Applications with Database and Application Macros

IMS online applications consist of individual programs scheduled to process transactions. (Scheduling must be invoked by JCL for batch message programs.) You must identify to the control program both the available programs and the transactions that they process. You must also identify the entire set of databases that can be referred to by the programs. Use the system definition macros in Table 3 to declare your applications.

If you want data sharing to be supported, you should first understand the interaction between the IMSCTRL and DATABASE macro statements and parameters.

Related Reading: Before coding your data-sharing system definition, read the information on administering a data-sharing environment in *IMS/ESA Administration Guide: System*.

Table 3. Database and Application Macros

Macro	Number Coded	Purpose
APPLCTN	1 per PSB	Names the application program that processes the transaction codes specified on the TRANSACT statements. The TRANSACT macro follows the APPLCTN macro in the Stage 1 input stream.
DATABASE	1 per HSAM, HISAM, or HDAM database; 2 per HIDAM database; 1 per secondary index database that refers to another database; 1 per HSDB, DEDB	Defines all databases that IMS online control program (and DBCTL environment) manages.
RTCODE	1 or more per APPLCTN statement	Specifies the routing codes that the input edit/routing exit routine uses to select a Fast Path application program.
TRANSACT	1 or more per APPLCTN statement; 0 for non-message BMP	Names the transaction codes to be processed by the application program specified on the APPLCTN statement. The APPLCTN macro physically precedes the TRANSACT macro in the Stage 1 input stream.

For a DBCTL environment, you define application programs only for BMP regions, using the APPLCTN macro. For an application program running in a CCTL region or using the ODBA interface, use the APPLCTN to define the PSB names that the applications require. The TRANSACT macro is not used in defining a DBCTL system.

Declaring Online Databases

The IMS control region allows the application program to access the database and provides DL/I call services. To declare online databases, you must define a list of all the physical databases that can be used by the programs. You do this by including DATABASE macro statements in the system definition stage 1 input. You must include separate statements for the index and data portions of a HIDAM database.

Defining Online Applications

Also include one statement for each secondary index database that refers to any other database defined to the online system.

The name you use on the DATABASE macro is the same as that on the DBD statement (or the member name in IMS.DBDLIB) for the corresponding physical or index database definition. Do not specify logical databases.

The DATABASE macro allows you to specify a performance option for an individual database. Specify RESIDENT to load the control data management block (DMB), which is needed to access the database, from the active IMS.ACBLIBA/B data set into a separate DMB pool at system initialization. The resulting advantage of this option is that these databases do not incur OPEN or CLOSE penalties caused by competition for space in the DMB pool.

The default value (EX) of the ACCESS keyword specifies that the database is for exclusive use by this online system.

Related Reading: If you plan to allow concurrent use by other IMS systems, refer to “Declaring Online Databases That Share Data” in *IMS/ESA Administration Guide: System* for an explanation of the other values for the ACCESS keyword.

Declaring Online Programs

The online system identifies an application by a unique Program Specification Block (PSB) name, which you code in the APPLCTN macro. The message processing program uses this same name for identification. Essentially, the PSB accesses the message input queue and declares any alternative destinations, other than the input source, for messages sent by the program. The major portion of the PSB defines a complete list of database access intentions (down to the segment level, or to the field within the segment). Before access, the PSB is prepared as a member of IMS.ACBLIB. The online system expects this control block to have been predefined.

Declaring Program Characteristics

The APPLCTN macro allows you to specify the characteristics for application programs that run under the control of DB/DC, DBCTL, and DCCTL systems.

You decide what type of application program is being defined, either batch message or message processing. You also specify the transaction class for the messages that the program is to receive, rather than code it on subsequent TRANSACT macros.

Requirement: The APPLCTN macro requires that you specify whether the program can be concurrently scheduled into more than one message region or batch message region. Indicating SCHDTYP=PARALLEL permits the program to schedule into more than one region; however, this option requires that processing be truly independent. Each schedule requires its own section of any shared pools. Program isolation activity can increase if the processing has the potential to perform updates in the same database record. You cannot control the ultimate processing sequence for transactions of the same type.

On the APPLCTN macro, you can also cause the scheduling process of all environments to generate a PSB containing an I/O PCB and an alternate modifiable PCB. This generated PSB can be used, for example, by an application that makes only SQL calls. PSBGEN and ACBGEN are not required when a GPSB is utilized. You can use the GPSB= parameter to generate this type of PSB.

Choosing PSB Performance Options

The RESIDENT option of the APPLCTN macro allows you to specify if the PSB is to be made resident during system initialization.

If the PSB is resident, the PSB is managed as follows:

- During system initialization, the total space for all resident PSBs is computed. This amount of space is obtained, and all resident PSBs are read from the active Application Control Block Library (ACBLIB) into this space.
- If SCHDTYP=PARALLEL when the program is scheduled, the resident PSB is allocated for that scheduling if it is not currently allocated to an executing program. If the resident PSB is being used, the PSB pool is searched for an inactive copy of the PSB. If one is found, that copy is allocated and used. If none is found, space for a copy is obtained in the PSB pool, and the resident PSB is copied into that space. The addresses within the copied PSB are then updated based on its new location, and its current status related to the former copy is reset.

If SCHDTYP=SERIAL when the program is scheduled, the resident copy of the PSB is allocated for that scheduling. Multiple copies of the PSB are not required; therefore, copies of the PSB are not required in the PSB pool.

Whether SCHDTYP=SERIAL or PARALLEL is specified, the PSB is read only once during initialization.

If the PSB is not resident, the PSB is managed as follows:

- When the program is scheduled, the PSB pool is searched for an inactive copy of the PSB and, if one is found, it is allocated and used. If none is found, space for a copy is obtained in the PSB pool.

If an active copy exists in the pool, it will be copied, its addresses are updated, and its status is reset to make it available for use. If no active copy exists, the PSB is read from the active ACBLIB.

When it is necessary to release a PSB in the pool, preference is given to retain “only copies” of PSBs to minimize ACBLIB I/O.

Use of the resident option eliminates I/O to ACBLIB when the PSB is scheduled. No storage fragmentation occurs in the resident PSB space as it does in the PSB pool, so more efficient use of storage is the result.

The use of the RESIDENT option should be based on the frequency with which the PSB is used. If the frequency of use would cause at least one copy of the PSB to normally be in the pool even if it were not defined as resident, it should be declared resident. If the PSB is used only occasionally, it should not be declared resident, so that its space in the PSB pool can be released when the PSB is not being used.

Using a Dynamic PSB

The specification of the dynamic PSB option, using the dynamic option (DOPT) parameter on the APPLCTN macro, causes the PSB to be read from the active ACBLIB each time the program is scheduled. This allows an ACBGEN for a PSB to be performed while the online IMS system is running. The result of that ACBGEN can be reflected in the next scheduling of the program using that PSB.

If DOPT is specified, the PSB is managed as follows:

- No initialization for the PSB is performed at system initialization. The PSB does not need to be in the active ACBLIB during system initialization.
- When the program is scheduled, the PSB is read from the ACBLIB into the PSB pool.

Defining Online Applications

- The space in the PSB pool is released when the program terminates.

DOPT can be applicable in a test configuration. It can also provide a PSB to be used with the Online Database Image Copy utility.

Restriction: The following restrictions apply to a DOPT PSB:

- A dynamic PSB cannot be made resident.
- A program using a dynamic PSB cannot be scheduled in parallel.
- An MPP scheduled against a dynamic PSB cannot go through Quick Reschedule or become a pseudo WFI.
- All databases referenced by the dynamic PSB must have been defined to the system and must be present in ACBLIB at system initialization or after the last online change was performed. This is because a corresponding option does not exist for DMBs, even though the PSB is dynamic.
- The current ACBLIB must consist of two or more concatenated data sets, and the dynamic PSB must reside in any data set in the concatenation other than the first. The concatenated data sets must be of the same format and contain the output from an ACBGEN.

Declaring Batch Message Programs

You must describe a BMP as BATCH on the PGMTYPE parameter of the APPLCTN macro. Include each BMP as a separate application with its own APPLCTN macro. You declare the BMP program by its use of a PSB, even though the batch JCL allows you to specify a program name that is different from the PSB name.

If a generalized BMP can execute with different PSBs, you must include APPLCTN macros for all of them. A choice of PSB usually involves several queues able to be processed by the one program. At execution, use the IN= parameter to match transaction queues to PSBs.

Defining IMS Transactions

Restriction: This section does not apply to DBCTL.

The TRANSACT macro influences the overall online IMS system response to incoming messages. Relate one or more transaction codes to the PSB named in the prior APPLCTN macro by using the TRANSACT macro to schedule the program for execution. The transaction characteristics declared in the TRANSACT macro are summarized in Table 4.

Table 4. Summary of Transaction Characteristics

Transaction Characteristic	TRANSACT Parameter	Parameter description
What are the transaction identifiers?	CODE SYSID	Identifies unique transaction codes. For Fast Path, also declares a possible routing code. Identifies remote system and local system.
What rules govern program scheduling for this transaction?	DCLWA PARLIM PRTY	Log write-ahead option. Queue count that triggers parallel scheduling. Normal priority and alternate priority when queues build up.
What exceptions are there to the scheduling rules?	MAXRGN PROCLIM SCHD	Limit for number of regions that can be scheduled per transaction. Limit for number of input messages processed or the processing time. Alternative options when the transaction is unavailable.

Table 4. Summary of Transaction Characteristics (continued)

Transaction Characteristic	TRANSACT Parameter	Parameter description
What type of processing?	INQ	Update database or not. Recovery required or not.
	FPATH	Fast Path potential transaction.
	MODE	Whether or not transactions are autonomous.
	MSGTYPE	Multisegment code or not.
	ROUTING	Response before more input or not.
	SERIAL	Whether or not program is aware of originating system.
	SPA	Serial processing of transaction.
What edit operations?	WFI	Conversational transaction. Wait for input transaction.
	EDIT	Whether or not to convert input data to uppercase. Name of an input edit routine.
What limits on output?	SEGNO	Maximum segments per transaction.
	SEGSIZE	Maximum segment size.

From the application, you need to determine how a transaction is to be processed and how it looks to the user. Table 5 helps you specify the correct parameters on the TRANSACT macro. For a more detailed explanation of this macro, see “TRANSACT Macro” on page 173.

Table 5. TRANSACT Macro. Transaction Configurations and Recommended Parameters

Configuration	Keyword Parameter
Declares that the program should be continuously available and not reloaded each time it is scheduled.	WFI
Specifies that the transaction is an inquiry, and its database processing does not cause any updates. No recovery at system restart is desired.	INQ=(YES,NORECOV)
Declares that the transaction is an inquiry, but update processing might occur.	INQ=(NO,RECOV)
Declares that a response to an input message is the next event.	MSGTYPE=(RESPONSE)
Declares that the response to an input message allows multiple message input actions.	MSGTYPE=(NONRESPONSE)
Specifies that database buffers are to be written to direct access upon each request for a new message. SNGL is forced for WFI applications.	MODE=SNGL
Specifies a group of transactions that are processed together.	MODE=MULT
Declares that the transaction is to be conversational. The scratchpad area size in bytes is given.	SPA= SPA=(size in bytes)
Declares that an input message edit routine is present. The member name of the link-edited module in IMS.USERLB (or equivalent library) is <i>name</i> parameter. UC specifies that input is translated to uppercase before presentation to program.	EDIT=(UC, <i>name</i>)
Protects output message queues by testing the size of an output segment.	SEGSIZE= <i>size</i>
Protects output message queues by testing the number of segments issued per transaction processed.	SEGNO= <i>number</i>

Application Program Output Limits

When an application program exceeds the output limits, a status code is returned, indicating an error. Any further attempt by the application program to exceed the limits results in abnormal termination of the program. To prevent abnormal terminations, specify values for the SEGSIZE and the SEGNO keywords on the TRANSACT macro between 0 and 65535. This causes the program to test the number and size of application program segments. This, in turn, helps prevent

Defining Online Applications

abnormal terminations that occur when application programs loop while inserting messages or segments into the message queues, or when they inadvertently insert segments of invalid lengths.

Defining Terminals with Data Communication Macros

Restriction: This section does not apply to DBCTL.

Most of the system definition stage 1 input is made up of declarations that define terminals to be attached to the IMS online system. Sixteen macro statements describe all of your IMS system's data communication facilities. Different combinations of these statements can be used to form four macro sets:

- BTAM, BSAM, GAM, and ARAM
- Switched Communication Devices
- Multiple Systems Coupling
- VTAM

If you are using the Extended Terminal Option (ETO) of IMS TM to define terminals, you can reduce the number of macro statements required for system-defined VTAM terminals. For more information on ETO, refer to "Including ETO in the IMS System" on page 23.

An overview of each macro set and a detailed description of the macro statements in each set follow in Table 6. Use this information and the configuration diagrams to define your terminals. Each terminal type and hardware option have equivalent parameters in one or more of the terminal-related macros in the stage 1 input.

In some instances, you must prepare the same macro statements for more than one of the sets. Refer to the section "Sequencing Stage 1 Input" on page 40 for information about the order of entering the macro sets into the assembler input stream. IMS issues stage 1 output warning messages and does not complete system definition if the macro sets are not entered in this order.

Defining Non-VTAM Terminals

If your IMS system uses BTAM, BSAM, GAM, or ARAM, use the macro statements from this set to describe your data communication facilities. The number of sets you must prepare depends upon the hardware configuration of your IMS system.

All non-VTAM data communication specifications must precede the VTAM macro set in your IMS system definition stage 1 input deck. You receive a stage 1 output warning message if the VTAM macro set is not the last physical set. If an MSC macro set is part of your system definition, it must follow your BTAM macro set and precede your VTAM macro set; otherwise, your system definition does not complete. To add VTAM support, you must specify either an ONLINE or ALL system definition on the IMSCTRL macro statement. Non-VTAM data communication macros appear in Table 6.

Table 6. Non-VTAM Data Communication Macros

Macro	Number Coded	Purpose
COMM ¹	1 only per IMS system definition	Specifies general communication options not associated with any particular terminal type

Table 6. Non-VTAM Data Communication Macros (continued)

Macro	Number Coded	Purpose
CTLUNIT	1 for each control unit attached to the line specified by the LINE statement	Provides terminal control unit address and attributes
LINE	1 for each line following the LINEGRP data	Assigns address and characteristics of one line in a line group
LINEGRP	1 for each line type	Assigns DD names for a group of lines and terminals with like attributes
NAME	1 or more for each logical terminal or component following each TERMINAL macro	Specifies the logical terminal name for the physical terminal specified by the TERMINAL macro
STATION	1 for each System/3 and System/7	Provides System/3 or System/7 station address and attributes
TERMINAL ²	1 for each physical terminal attached to the line specified by the LINE statement	Specifies physical terminal characteristics

Note:

1. You can specify only one COMM macro in an IMS system definition. COMM is optional for the BTAM macro set but is required for VTAM.
2. When the TERMINAL statement describes a switched physical terminal, the NAME statement cannot be used to specify logical terminal names for it.

Specifying the Non-VTAM Terminal Configuration

An example of coding for two 3270 local line groups is shown below:

Example:

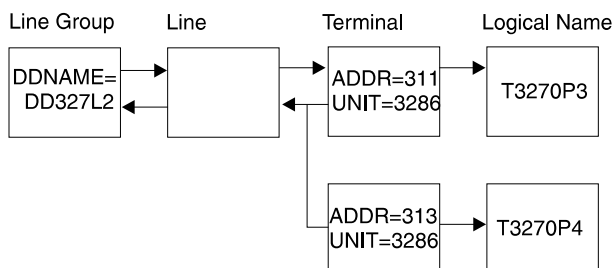
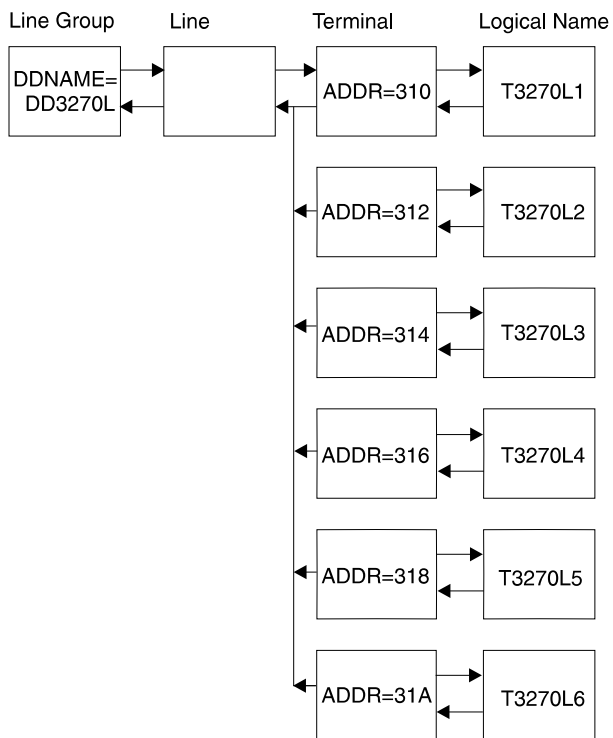
```

LINEGRP DDNAME=DD3270L,UNITYPE=(3270,LOCAL)
LINE BUFSIZE=300
  TERMINAL ADDR=310,MODEL=2
  NAME T3270L1
  TERMINAL ADDR=312,FEAT=(PFK,CARD,PEN),X
    OPTIONS=FORCRESP,MODEL=1
  NAME T3270L2
  TERMINAL ADDR=314,TYPE=3270-A1,SIZE=(12,80)
  NAME T3270L3
  TERMINAL ADDR=316,TYPE=3270-A2,SIZE=(24,80)
  NAME T3270L4
  TERMINAL ADDR=318,TYPE=3270-A5,SIZE=(12,40)
  NAME T3270L5
  TERMINAL ADDR=31A,TYPE=3270-A6,SIZE=(6,40)
  NAME T3270L6
LINEGRP DDNAME=DD3270L2,UNITYPE=(3270,LOCAL)
LINE
  TERMINAL ADDR=311,MODEL=2,UNIT=3286
  NAME T3270P3
  TERMINAL ADDR=313,MODEL=1,UNIT=3286
  NAME T3270P4

```

The configuration for the 3270 local line group is shown below:

Defining Terminals



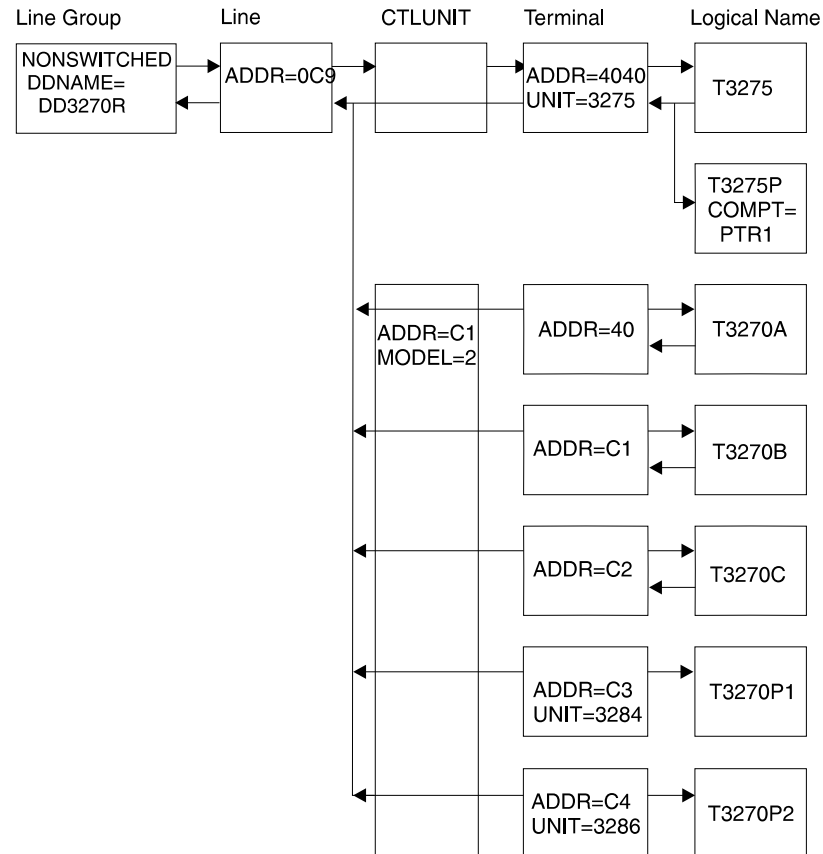
Following is an example of coding for a 3270 remote line group:

Example:

```

LINEGRP DDNAME=DD3270R,UNITYTYPE=3270, CODE=EBCDIC
LINE ADDR=0C9
  TERMINAL ADDR=4040,UNIT=3275,COMPT=PTR1,          X
    FEAT=PFK,OPTIONS=COPY
  NAME T3275
  NAME T3275P,COMPT=PTR1
  CTLUNIT ADDR=C1,MODEL=2
  TERMINAL ADDR=40,FEAT=(PFK,CARD,PEN),            X
    MODEL=2,OPTIONS=COPY
  NAME T3270A
  TERMINAL ADDR=C1,FEAT=(PFK,CARD,PEN),            X
    MODEL=2,OPTIONS=COPY
  NAME T3270B
  TERMINAL ADDR=C2,FEAT=(PFK,CARD,PEN),            X
    MODEL=1,OPTIONS=COPY
  NAME T3270C
  TERMINAL ADDR=C3,UNIT=3284,MODEL=2,BUFSIZE=132
  NAME T3270P1
  TERMINAL ADDR=C4,UNIT=3286,MODEL=2,BUFSIZE=132
  NAME T3270P2
  
```

The configuration for the 3270 remote line group is shown below:



Defining Intelligent Stations

If you plan to include remote intelligent stations in the IMS network, specify the station's physical and logical characteristics on a STATION macro. Set aside one or more line groups, and then add the characteristics of one or more System/3 or System/7 stations. An LTERM name identifies a station. IMS assigns default LTERM names as it encounters STATION macros—RSTSnnnn, where nnnn is a sequence number incremented from 0001.

The LINEGRP macro allows you to declare a System/7 line to operate either as a polled line or in contention mode. Record a two-digit hexadecimal number as the physical address of a System/7 operating on a polled line. Also declare whether the transmission is to be binary synchronous or start-stop. Optional characteristics are:

- Transmission data format (not for System/3 stations)
- Alignment and padding of data segments
- Maximum number of output blocks to the station
- No output is transmitted when the station is started or output is queued until the station issues an ASK command

Terminals attached to a System/3 or System/7 can transmit messages to IMS; however, the transmission block formats and protocols must coincide with those defined for IMS intelligent remote station support (IRSS). Each terminal can be restricted to input data only. Conversational processing and preset output destination are available.

Defining Terminals

Related Reading: For details on how to communicate between IMS and IRSS, refer to *IMS/ESA Administration Guide: Transaction Manager*.

Defining Switched Communication Devices

If you plan to use switched 3275 devices, use the macro statements from Table 7 set to define your terminals. If prepared, these statements must be included as a subset of those prepared for the LINEGRP macro statement. Define a line group and follow this data with details of the hardware options indicated by CONFIG macro keywords. Each 3275 device carries a physical identification number as part of its circuitry. This number is made up of the last three bytes of the terminal's hardware identification and the label on the CONFIG macro. List all authorized identifications for a line group on an IDLIST macro. Subsequent LINE macros coded for 3275 devices can point to these lists. The number of sets you must prepare depends upon the hardware configuration of your IMS system. Table 7 lists macros for defining switched communication devices.

Table 7. Switched Communication Device Macros

Macro	Number Coded	Purpose
CONFIG	One for each type of switched 3275 configuration	Describes the configuration for a switched 3275 terminal
IDLIST	One for each switched device; must follow the LINE macro	Provides a list of hardware identifications for switched 3275 devices for a line group
LINE	One for each line of the line group	Provides the address and characteristics of one line in the line group specified by the LINEGRP statement
LINEGRP	One for each line type	Assigns DD names for a group of lines and terminals with like attributes
NAME	One or more for each logical terminal or component; must follow the SUBPOOL statement	Provides logical terminal names
POOL	One for each switched line group or one for all switched lines; must be followed by a SUBPOOL macro	Defines a pool of logical terminal names
SUBPOOL	One or more for each subpool; must follow the POOL macro	Defines a subpool of logical terminal names
TERMINAL	One for each physical terminal attached to the line specified by the LINE statement	Provides physical terminal data

Specifying Macros for Multiple Systems Coupling

If you plan to use Multiple Systems Coupling (MSC), you must prepare one set of MSC macro statements to describe the additional data communication facilities.

Although Intersystem Communication (ISC) is part of MSC, the macro statements used to define ISC are included in the VTAM macro set and not in the MSC macro set.

Defining Terminals

Restriction: You must enter the macro sets in the order specified in “Sequencing Stage 1 Input” on page 40 of your IMS system stage 1 input deck. IMS issues stage 1 output warning messages and does not complete system definition if the macro sets are not entered in this order.

When you use MSC in your IMS system, the BTAM, BSAM, GAM, ARAM, and switched device macro sets (if used) must precede the MSC macro set in your IMS system definition stage 1 input deck. The VTAM macro set (if used) must occur after the MSC macro set in the stage 1 input deck. Table 8 lists macros for Multiple Systems Coupling (MSC).

Table 8. MSC Data Communication Macros

Macro	Number Coded	Purpose
MSLINK	One for each logical system-to-system link	Defines a logical link to another system
MSNAME	One for each remote and local system identification	Defines the link name block that relates to the MSLINK statement and the corresponding remote and local system identifications
MSPLINK	One for each physical link	Defines a physical link to another system in an IMS MSC configuration
NAME	One or more for each logical terminal that is to be referenced by your IMS system	Provides the logical terminal name for a physical terminal that is part of the remote system specified in the MSNAME statement

Related Reading: The attributes specified for MSC are described in *IMS/ESA Administration Guide: Transaction Manager*.

Defining VTAM Terminals

If VTAM is used by your IMS system, prepare a set of VTAM macro statements to describe the VTAM data communication facilities. The number of sets you must prepare depends upon the hardware configuration of your IMS system.

To add VTAM support, an ONLINE or ALL system definition must be specified on the IMSCTRL macro statement.

For information on the ETO options available for defining dynamic VTAM terminals, refer to the ETOFEAT keyword in the IMSCTRL macro. on page You must enter the macro sets in the order specified in “Sequencing Stage 1 Input” on page 40 your IMS system stage 1 input deck. IMS issues stage 1 output warning messages and does not complete system definition if the macro sets are not entered in this order.

All non-VTAM data communication specifications must precede the VTAM macro set in your IMS system definition stage 1 input deck; a stage 1 output warning message occurs if the VTAM macro set is not the last physical set. If an MSC macro set is part of your system definition, it must precede the VTAM macro set, or your system definition does not complete. Table 9 on page 20 lists macros for defining VTAM terminals.

Defining Terminals

Table 9. VTAM Data Communication Macros

Macro	Number Coded	Purpose
COMM ¹	One only per IMS system definition	Specifies general communication options not associated with any particular terminal type
NAME	One or more for each physical terminal or component	Specifies the logical terminal name for physical terminals of the type specified by the TERMINAL or SUBPOOL statement
SUBPOOL	One for each dynamically allocated session	Defines logical unit type 6 (LU 6.1) subpools that can be dynamically allocated within the VTAMPOOL
TERMINAL	One for each terminal or component; must follow the TYPE macro	Provides the characteristics of the physical terminal type specified in the TYPE statement
TYPE	One for each terminal type	Specifies a group of VTAM terminals of the same type
VTAMPOOL	One or more per SUBPOOL; the only purpose for multiple VTAMPOOL statements is to facilitate the documentation of the intended use of the logical terminal group	Defines subpools of logical terminals to be dynamically allocated to LU 6.1 sessions

Note: ¹ You can specify only one COMM macro in an IMS system definition. COMM is required for the VTAM macro set.

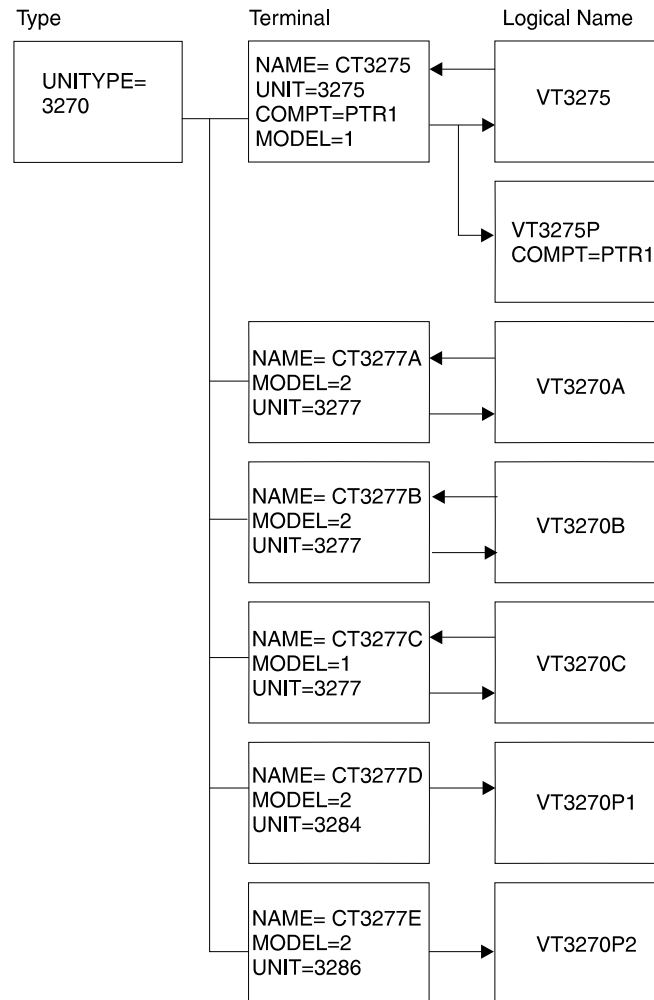
Specifying the VTAM Terminal Configuration

An example of coding for a group of 3270 VTAM terminals is shown below:

Example:

```
TYPE UNITYPE=3270,MODEL=2,PTRSIZE=132,OPTIONS=COPY
TERMINAL NAME=CT3275,UNIT=3275,COMPT=PTR1,MODEL=1
NAME VT3275
NAME VT3275P,COMPT=PTR1
TERMINAL NAME=CT3277A
NAME VT3270A
TERMINAL NAME=CT3277B
NAME VT3270B
TERMINAL NAME=CT3277C,MODEL=1
NAME VT3270C
TERMINAL NAME=CT3277D,UNIT=3284
NAME VT3270P1
TERMINAL NAME=CT3277E,UNIT=3286
NAME VT3270P2
```

The configuration of 3270 VTAM terminals that these macros describe follows:



Specifying the Master Terminal

The master terminal is the key control point for IMS online operations. Choosing a video device as the master terminal provides the advantage of convenient data entry and output response. Keep in mind, however, that you need a printed copy of many of the responses to commands, as well as a record of the system messages sent to the master terminal.

To indicate your choice for the master terminal, specify MASTER after the logical terminal name on the NAME macro. The NAME macro must follow the TERMINAL macro that describes the terminal chosen. Similarly, specify SECONDARY after the logical terminal name on the NAME macro to indicate your choice for secondary master terminal.

Restriction: Primary and secondary master terminals cannot be ETO or LU 6.2 device terminals. See "Member DFSDCxxx" on page 305 for additional restrictions that apply when you specify the DFSDCxxx PROCLIB member.

Choosing Master Terminal Devices

Some restrictions apply for choosing the device for master and secondary terminals, as shown in Table 10 on page 22. Your choice for secondary master terminal depends on the expected amount of output and the user's requirement for

Defining Terminals

promptness of printing.

Table 10. Device Choices for Master Terminals

Master Terminal	Choice of Device Type
Primary	3270 ¹ SLUTYPE1 (console) SLUTYPE2 (console) ² 2740 ³
Secondary	328x ¹ SLUTYPE1 (first component) 2740 ³ SPOOL

Notes:

¹ If a 3270 is specified as the master terminal, a 328x or SPOOL device must be specified as secondary.

² SLUTYPE2 requires a secondary master terminal. The NAME macro for the secondary must be placed before of the primary terminal.

³ The 2740 must have station control.

The /ASSIGN command can be used to switch the secondary master console to another destination such as a spool SYSOUT line group.

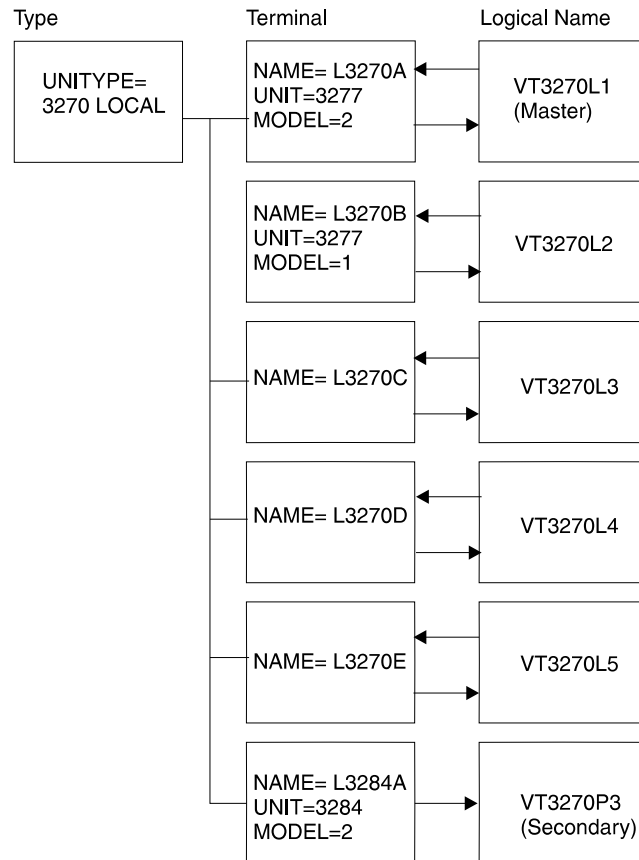
Specifying the Master Terminal Configuration

An example of coding for a master terminal on a 3270 local line is shown below:

Example:

```
TYPE  UNITYPE=(3270,LOCAL),MODEL=2
TERMINAL  NAME=L3270A
NAME  (VT3270L1,MASTER)
TERMINAL  NAME=L3270B,MODEL=1,OPTIONS=FORCRESP
NAME  VT3270L2
TERMINAL  NAME=L3270C,TYPE=3270-A2,SIZE=(24,80)
NAME  VT3270L3
TERMINAL  NAME=L3270D,TYPE=3270-A3,SIZE=(32,80)
NAME  VT3270L4
TERMINAL  NAME=L3270E,TYPE=3270-A4,SIZE=(43,80)
NAME  VT3270L5
TERMINAL  NAME=L3284A,UNIT=3284,PTRSIZE=132
NAME  (VT3270P3,SECONDARY)
```

The configuration for the master terminal is shown below:



Choosing the Extent of Secondary Master Logging

To provide for automatic copying of the entry and response of key system control commands, specify values for the COPYLOG keyword on the COMM macro. These values cause all command activity issued by any terminal to be copied.

Recommendation: We recommend that you specify ALL, unless terminals other than the master terminal are likely to issue many commands. The choice of NONE or NOMASTER is not recommended because the printed log of activity provides a valuable audit mechanism. Your choice of automatic copying does not affect the logging of IMS system messages to the secondary master terminal.

The commands that can be copied to the secondary master terminal are provided under the COMM macro description.

Including ETO in the IMS System

The Extended Terminal Option (ETO) of IMS TM enables VTAM terminals to logon to IMS TM, even if they are not defined during the IMS system generation process. IMS TM dynamically builds the required control blocks and queues based on VTAM information and IMS skeleton definitions called descriptors. ETO is an optional feature defined by the IMSCTRL macro statement.

Definitions: Terminals defined with system generation procedures are called *static* terminals. Terminals not defined with system generation procedures are called *dynamic* terminals.

Including ETO

The ETO feature applies to all VTAM terminals except for MSC VTAM, MTO terminals, and XRF surveillance links. These, and terminals supported by non-VTAM access methods, still require an IMS generation to introduce changes. The number of terminals that you define dynamically is limited only by your resource constraints. With ETO, you can:

- Improve your system availability by reducing scheduled down time associated with adding or deleting VTAM terminals
- Improve IMS security by relating output message queues (LTERMs) to users rather than to terminals
- Reduce the number of macro statements required for static, system-defined VTAM terminals
- Reduce virtual storage use for ETO terminals and users by allocating storage only when it is actually required

Restriction: Without ETO, adding, deleting, or changing terminals defined to IMS TM requires that your online system be terminated down to incorporate the changes. Additionally, the stage 1 input can become large if you are defining a large number of terminals.

With ETO, IMS TM does not need to be terminated and cold started to add new terminals and users. ETO descriptors are not generated during LGEN; however, a standard system generation can build ETO descriptors that support terminals that are currently generated statically. This allows an easy transition to the ETO environment.

Dynamic allocation of VTAM terminals does not require the use of VTAM TYPE, TERMINAL, NAME, VTAMPOOL, or SUBPOOL macros, or the use of MSC remote NAME macros, for the system definition stage 1 input. Instead, you include ETO with the ETOFEAT keyword on the IMSCTRL macro, which causes IMS TM to generate descriptors, as mentioned above. IMS TM uses descriptors to build the required control blocks and queues associated with terminal definition. With ETO terminals, configuration information from VTAM control blocks is dynamically merged with information from ETO descriptors when a user logs onto IMS TM. Terminal control blocks are not built for ETO terminals until an ACF/VTAM session is established between the terminal and IMS TM, or until a user structure is built. User control blocks are not built until a user signs on to a terminal.

You can continue to define ACF/VTAM terminals statically.

Requirement: Static terminal definition is still required for:

- BTAM (non-VTAM) terminals
- MTO and secondary master terminals
- MSC physical/logical links
- XRF ISC Surveillance link

ETO descriptors are described in “Enabling ETO Support for ACF/VTAM Terminals” on page 399.

Related Reading: For information on administering ETO, see *IMS/ESA Administration Guide: Transaction Manager*. For information on security for ETO, see *IMS/ESA Administration Guide: System*.

Including FDBR in the IMS System

DB/DC

To enable a DB/DC subsystem for Fast DB Recovery (FDBR), you specify the FDRMBR parameter in the IMS procedure. The FDRMBR parameter defines the DB/DC system as FDBR-capable.

Restriction: If both FDRMBR and HSBID (XRF configuration) parameters are specified in the DBC procedure, the FDRMBR parameter is ignored.

You must supply the IMS ID of the system that FDBR is to track. You specify this IMS ID in the control statement for the DFSFDRxx IMS.PROCLIB member, and it must match the ID you specified for the IMSID EXEC parameter.

DBCTL

To enable a DBCTL subsystem for Fast DB Recovery, you specify the FDRMBR parameter in the DBC procedure. The FDRMBR parameter defines the DBCTL system as FDBR-capable. You should make note of the IMS ID of the system FDBR will track. You specify this IMS ID in the control statement for the DFSFDRxx IMS.PROCLIB member, and it must match the ID you specify for the IMSID EXEC parameter.

Restriction: If you specify both FDRMBR and DBRSE (DBCTL standby configuration) parameters in the DBC procedure, the FDRMBR parameter is ignored.

Including RSR in the IMS System

The system definition process for Remote Site Recovery (RSR) involves setting up a global service group (GSG) and a transport manager instance (TMI) both of which can be defined in several different places. This section describes how the various parts of RSR system definition fit together.

You must specify RSRFEAT=RLT or RSRFEAT=DLT in your IMSCTRL macro to enable RSR. You also need to specify a GSG name and a TMI name.

Additional RSR-related parameters and overrides are specified in a DFSRSRxx member that you must supply in IMS.PROCLIB.

As Figure 1 on page 26 shows, you can specify a GSG name in several places: the IMSCTRL macro, the DLIBATCH procedure, the DBBBATCH procedure, the DFSRSRxx PROCLIB member, and on certain DBRC commands. Figure 2 on page 27 shows where you can specify a TMI name: the IMSCTRL macro, the DLIBATCH and DBBBATCH procedures, the DFSRSRxx PROCLIB member, and also on certain transport manager and isolated log sender commands.

To define defaults for the GSG or TMI names, specify the GSGNAME= or TMINAME= statement in the IMSCTRL macro. To override the specification in the IMSCTRL macro, specify the:

- GSGNAME= and TMINAME= statements in the DLIBATCH or DBBBATCH procedure
- GSGNAME and TMINAME parameters in the DFSRSRxx PROCLIB member

Including RSR

After the GSG name and TMI name have been set or overridden, you can use these names in various DBRC and transport manager and isolated log sender commands.

Example: If your GSG name is payroll for your payroll system and your TMI name is ELX1, these are the DBRC, TMS, and ILS commands:

- DBRC commands:

```
INIT.GSG GSGNAME(payroll)
INIT.DB DBD(db1) GSGNAME(payroll)
CHANGE.DB DBD(db1) GSGNAME(payroll)
INIT.DBDS DBD(dedb1) AREA(area1) GSGNAME(payroll)
CHANGE.DBDS DBD(dedb1) AREA(area1) GSGNAME(payroll)
```

- TMS and ILS commands

```
SET INSTANCE(ELX1)
START ILS(payroll)
```

In Figure 1, payroll is an example of a GSG name. The broken line represents the point at which the default or overridden GSG name can be used.

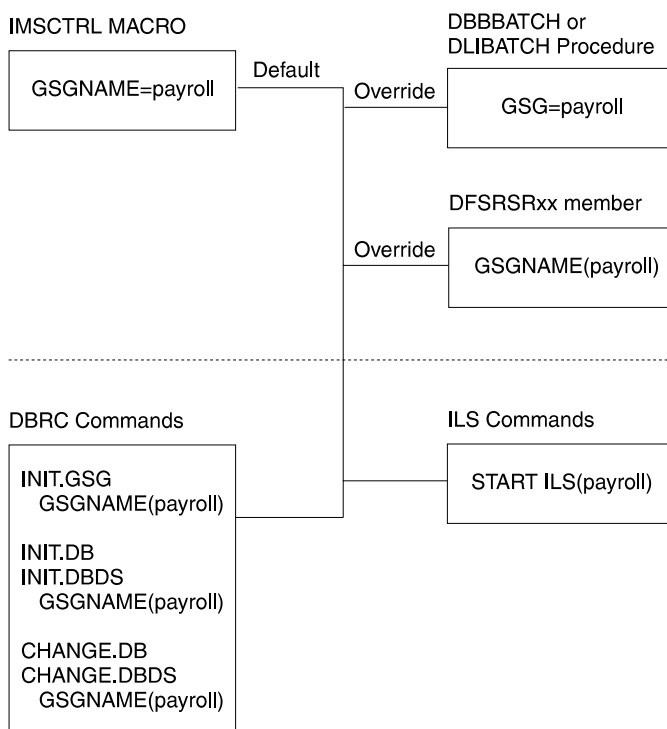


Figure 1. Specifying the Global Service Group (GSG) Name for RSR

In Figure 2 on page 27, ELX1 is an example of a TMI name. The broken line represents the point at which the default or overridden TMI name can be used.

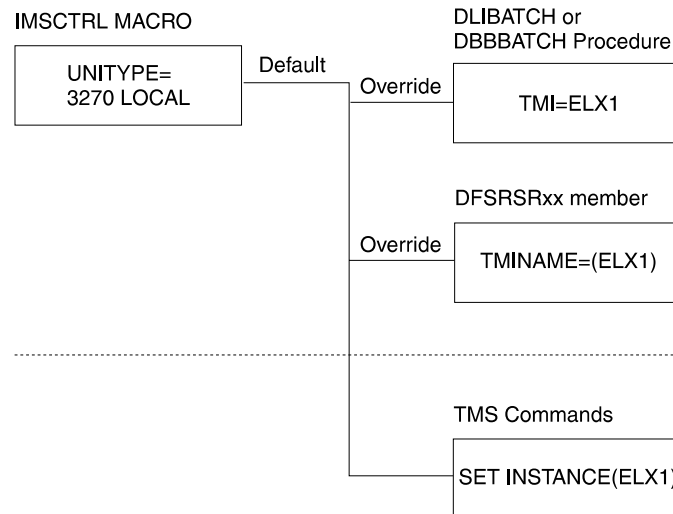


Figure 2. Specifying the Transport Manager Instance (TMI) Name for RSR

A single active online subsystem environment does not require a separate system generation for the tracking subsystem. A separate tracking subsystem system generation is required if:

- More than one active subsystem must be tracked, and no one active subsystem system generation defines all of the resources needed by the tracker.
This separate system generation requirement can be eliminated if the system generation of one of the active subsystems is enlarged to define the additional resources.
- Batch DL/I jobs use databases that are not defined in the active subsystem.
- You want to eliminate database definitions for databases that are not tracked from the tracking subsystem.
- You want to reduce the data communications definitions for the tracking subsystem.

Including RSR

Chapter 2. The System Definition Process

IMS Version 6 is the first version of IMS in which the system generation process does not build a complete system. Some elements are built outside of the system generation process. Member DFSJCLIN in IMS.DFSISRCA contains the JCL necessary to build these elements.

Important: These elements are built during IMS INSTALL/IVP with jobs IV_B340J and IV_B350J.

In this Chapter:

“Selecting the Appropriate IMS System Definition”

“Managing System Definition” on page 40

“Verifying System Definition” on page 41

“Running System Definition Stage 1” on page 47

“Running System Definition Stage 2” on page 49

“Maintaining Stage 1 Output with SMP/E” on page 50

System definition examples for a variety of IMS environments are provided in “Appendix A. IMS System Definition (Step 3) Examples” on page 427. This appendix also provides examples of data communication macro statements and data sharing system configurations.

Selecting the Appropriate IMS System Definition

The IMS system definition includes CLASS and TYPE. The types are: ALL, BATCH, CTLBLKS, MODBLKS, MSVERIFY, NUCLEUS, and ON-LINE. The three system definition environment classes are: DB/DC, DBCTL, and DCCTL. DB/DC is the system definition class that builds a standard IMS system. The DBCTL class builds only the DBCTL environment. The DCCTL class builds only the DCCTL environment.

Table 11 describes the types of system definition specified by the IMSCTRL macro, their typical use, and the results.

Table 11. Types of System Definition

IMSCTRL Option	When Used	Result
ALL	Typical initial generation. Not usually needed for maintenance.	Builds most IMS libraries. Includes BATCH and ONLINE options.
BATCH	Only for the batch environment.	Builds batch environment libraries.
CTLBLKS	Convenience update. Includes link-edit of existing nucleus with same suffix.	Control block generation. Includes MSVERIFY and MODBLKS options.
MODBLKS	Convenience update. Used for the online change option.	Builds control blocks that enable database, program, transaction, and Fast Path routing code changes to be cut over during online operation.
MSVERIFY	Only appropriate for MSC.	Builds control blocks for the MSC Verification utility.
NUCLEUS	Major maintenance.	Builds IMS nucleus and control blocks. Includes CTLBLKS option.
ON-LINE	Major update or initial generation. Often required for maintenance.	Builds most IMS libraries. Includes all but BATCH option.

Selecting Appropriate System Definition

Requirement: After your initial system definition, the ON-LINE, CTLBLKS, and NUCLEUS types of generation are used to implement most changes. These generations require a cold start of the IMS online system to take effect.

However, for certain modifications and additions, you can take advantage of the online change method using the MODBLKS generation. The changes are made active during the execution of the online system and do not require a restart operation.

Related Reading: For more information on managing online change, refer to *IMS/ESA Administration Guide: System*.

Using Large System Generation

LGEN is a system definition configuration, not a system definition type. Use this configuration when stage 1 or stage 2 job steps require more than 4 megabytes of private storage for execution. You can specify LGEN as the fourth parameter of the SYSTEM= keyword for the IMSCTRL macro. For information on how to specify LGEN on IMSCTRL, see "IMSCTRL Macro" on page 82. ETO descriptors are not generated during LGEN.

IMS LGEN does not support the ICTL assembler instruction.

Related Reading: For an overview of the LGEN configuration, refer to *IMS/ESA Administration Guide: System*.

Using DBCTL

In this book, the term "online" refers to both a DB/DC and DBCTL environment. The term "DBCTL" refers to a system generated specifically for DBCTL, not to a DB/DC environment performing DBCTL services. Table 12 on page 32 specifies the macros that apply to DBCTL; use this information to identify which macros apply to which IMS configurations.

Any reference to LSO=S refers to the DLISAS in a DBCTL environment. Any reference to the MTO also applies to the system operator for DBCTL. References to IMS commands (for example, /START) apply to DBCTL, assuming that the "/" is the DBCTL command recognition character. And, any references to MFS are **not** applicable to a DBCTL environment. All other information applies to DBCTL unless specifically stated otherwise.

The minimum generation required to generate DBCTL is an ON-LINE type and a DBCTL class; for further information see the SYSTEM parameter of the IMSCTRL macro.

To define a batch nucleus within a DBCTL RESLIB, use BATCH as the type and DBCTL as the class. Use DBCTL class to run BATCH jobs and to process network transactions without IMS TM. This nucleus is used in the batch DL/I environment.

Using DCCTL

The term "DCCTL" refers to a system generated specifically for DCCTL, which is also an online environment. The system definition process for DCCTL is identical to the process required to define and generate an IMS DB/DC system. You can specify DCCTL as the fourth parameter of the SYSTEM= keyword for the IMSCTRL

Selecting Appropriate System Definition

macro. This causes the appropriate DCCTL environment to be generated. Table 12 on page 32 specifies the macros that apply to DCCTL.

No separate DL/I address space exists in a DCCTL environment.

To define a batch nucleus within a DCCTL RESLIB, use BATCH as the type and DCCTL as the class. This produces a TM batch environment, which can be used to access GSAM or DB2 data. Although the DCCTL environment does not support DL/I databases, DEDBs, or MSDBs, it does support GSAM databases. This support allows applications to issue symbolic checkpoint (CHKP) and extended restart (XRST) calls against the data set to allow data repositioning.

Using the Macro Table

Table 12 on page 32 shows, in order of complexity, which IMS system definition parameters can be modified by type of system definition. You can override some keywords for some macros by specifying certain keyword parameters on the EXEC statement for the IMS, DBC, or DCC procedure, or by modifying the JCL. Other keywords can also be overridden through IMS operator commands. For further information, see “Chapter 7. Tailoring the IMS System to Your Environment” on page 299.

Certain macro statements, keywords, and parameters require additional consideration if they are being newly added to or entirely deleted from the system. These are explained in the notes that follow Table 12 on page 32.

To use Table 12 on page 32, find the macro-associated keyword and parameter that you want to change. Then scan across the page to find the first column with an “X” under the system definition types. This identifies the minimum system definition required to change the parameter. If multiple changes are to be made, scan each macro statement on the figure to determine the minimum system definition. When changing operands, make certain the minimum system definition required to update the operand is performed. If you specify a lower level than that indicated in Table 12 on page 32 the results are unpredictable.

The JCL column in the table identifies the EXEC parameter, DFSPBIMS keyword, DFSPBDBC keyword, DFSPBDCC keyword, or IMS, DBC, or DCC operator command to override the operand in the macro.

Keywords and parameters changed during a MODBLKS system definition can be brought online through a series of /MODIFY operator commands, except as noted in Note 1 on page 38.

The CTLBLKS generation option on the IMSCTRL macro statement can be used **only** to replace the control blocks of an existing nucleus—that is, a nucleus having the same suffix.

Adding new device support features or options might require a NUCLEUS generation, because additional modules (for example, the VTAM terminal COPY option and module DFSCVEQ0) might need to be linked into the nucleus.

Some of the values can be displayed using the /DISPLAY command shown in table notes. The value in the /DISPLAY column is used to qualify the /DISPLAY command as necessary.

Table 12. Macro Table - Selecting the Appropriate IMS System Definition

Macro	Operand	Value	JCL or Command	/DISPLAY	MODBLKS	CTLBLKS	NUCLEUS	ON-LINE	ALL	MSVERIFY	DB BATCH	TM BATCH	DBCTL	DB/DC	DCCTL	Refer to Table Notes at the end of this table	
APPLCTN	DOPT				X	X	X	X	X				X	X	X		
	PGMTYPE	class	/ASSIGN		X	X	X	X	X					X	X	26	
		All others				X	X	X	X	X				X	X	X	
	RESIDENT				X	X	X	X	X				X	X	X		
	FPATH				X	X	X	X	X					X	X		
	SYSID					X	X	X	X	X					X	X	1
		All Others				X	X	X	X	X				X	X	X	
BUFPOOLS	DMB		DMB=	DMBP		X	X	X	X				X	X		2, 27	
	EPCB		EPCB=	EPCB		X	X	X	X				X	X	X	2, 27	
	FORMAT	size 1	FBP=	MFP		X	X	X	X					X	X	2, 27	
	FRE		FRE=			X	X	X	X					X	X	2	
	PSB		PSB=	PSBP		X	X	X	X				X	X	X	2, 27	
	PSBW		PSBW=	PSBW			X	X	X	X				X	X	X	2, 27
		SASPSB	size 1	CSAPSB=			X	X	X	X				X	X		2
		size 2	DLIPSB=			X	X	X	X				X	X		2	
COMM	COPYLOG		/SMCOPY			X	X	X	X					X	X		
	OPTIONS	NOFMAS FMTMAS					X	X	X					X	X		
		NOMFSTEST MFSTEST						X	X	X					X	X	2
		PAGING NOPAGE					X	X	X					X	X		
	NOMSPEX MSPEXIT					X	X	X	X				X	X	5		

Table 12. Macro Table - Selecting the Appropriate IMS System Definition (continued)

Macro	Operand	Value	JCL or Command	/DISPLAY	MODBLKS	CTLBLKS	NUCLEUS	ON-LINE	ALL	MSVERIFY	DB BATCH	TM BATCH	DBCTL	DB/DC	DCCTL	Refer to Table Notes at the end of this table
		NOUSEMSG USERMSG				X	X	X	X					X	X	5
		All Others				X	X	X	X					X	X	
	RECANY					X	X	X	X					X	X	3, 4
	All Others					X	X	X	X					X	X	
CONFIG						X	X	X	X					X	X	
CTLUNIT						X	X	X	X					X	X	
DATABASE	ACCESS		DD		X	X	X	X	X				X	X		1, 6, 7
			/START		X	X	X	X	X				X	X		1, 28
	RESIDENT				X	X	X	X	X				X	X		
	All Others				X	X	X		X					X		
FPCTRL								X	X				X	X	X	35
	BFALLOC	parm 1	DBFX=			X	X	X	X				X	X		2, 8
		parm 2	DBBF=			X	X	X	X				X	X		2, 8
		parm 3	BSIZ=			X	X	X	X				X	X		2, 8
	OThread		OTHR=			X	X	X	X				X	X		2, 8
	USRHASH		UHASH=			X	X	X	X				X	X		2, 8
	All Others					X	X	X	X				X	X		
IDLIST						X	X	X	X					X	X	
IMSCTF	APNDG							X	X				X	X	X	
	CPLOG					X	X	X	X				X	X	X	
	LOG					X	X	X	X			X	X	X	X	
	PRDR		PRDR=			X	X	X	X				X	X	X	2
	RDS					X	X	X	X				X	X	X	
	SVCNO	type 2						X	X		X	X	X	X	X	9

Table 12. Macro Table - Selecting the Appropriate IMS System Definition (continued)

Macro	Operand	Value	JCL or Command	/DISPLAY	MODBLKS	CTLBLKS	NUCLEUS	ON-LINE	ALL	MSVERIFY	DB BATCH	TM BATCH	DBCTL	DB/DC	DCCTL	Refer to Table Notes at the end of this table
		type 4						X	X				X	X	X	
IMSCTRL	CMDCHAR		CRC=			X	X	X	X				X			
	DBRC	parm 1	DBRC=						X		X	X	X	X	X	2, 10, 21
		parm 2	DBRC=						X		X	X	X	X	X	10, 21
	DBRCNM		DBRCNM=			X	X	X	X				X	X	X	2, 29
	DCLWA					X	X	X	X					X	X	
	DESC					X	X	X	X			X	X	X	X	
	DLINM		DLINM=			X	X	X	X			X	X	X		2, 29
	ETOFEAT							X	X					X	X	
	GSGNAME		GSGNAME=					X	X		X	X	X	X	X	36
	HSB							X	X					X	X	12
	IMSID		IMSID=			X	X	X	X			X	X	X	X	2
	IRLM		IRLM=			X	X	X	X			X	X	X		2, 11, 37
	IRLMNM		IRLMNM=			X	X	X	X			X	X	X		2, 11, 37
	MAXCLAS					X	X	X	X			X		X	X	
	MAXREGN		PST=			X	X	X	X				X	X	X	2
	MCS					X	X	X	X			X	X	X	X	
	MSVID					X	X	X	X	X			X	X	X	
	NAMECHK					X	X	X	X			X	X	X	X	
	RSRFEAT							X	X		X	X	X	X	X	
	SYSTEM							X	X		X	X	X	X	X	
	TMINAME		TMINAME=					X	X		X	X	X	X	X	36
IMSGEN	MACLIB				X	X	X	X	X		X	X	X	X	X	
	MFSDFMT					X	X	X	X					X	X	
	MFSTEST						X	X	X					X	X	2

Table 12. Macro Table - Selecting the Appropriate IMS System Definition (continued)

Macro	Operand	Value	JCL or Command	/DISPLAY	MODBLKS	CTLBLKS	NUCLEUS	ON-LINE	ALL	MSVERIFY	DB BATCH	TM BATCH	DBCTL	DB/DC	DCCTL	Refer to Table Notes at the end of this table
	PAGE						X	X	X					X	X	
	PSWDSEC					X	X	X	X				X	X	X	
	SECCNT					X	X	X	X				X	X	X	
	SUFFIX						X	X	X				X	X	X	
	SURVEY								X				X	X		13
	SYMSMSG					X	X	X	X					X	X	
	TERMSEC					X	X	X	X					X	X	
	All Others				X	X	X	X	X	X	X	X	X	X	X	
LINE	ADDR		DD			X	X	X	X					X	X	14, 30
	All Others					X	X	X	X					X	X	
LINEGRP						X	X	X	X					X	X	15
MSGQUEUE	BUFFERS	NBR	QBUF=			X	X	X	X					X	X	2
		SIZE				X	X	X	X					X	X	
	All Others					X	X	X	X					X	X	
MSLINK						X	X	X	X					X	X	
MSNAME						X	X	X	X					X	X	
MSPLINK	ADDR		DD			X	X	X	X					X	X	14, 16, 31
	MODETBL		/CHANGE	link# MODE		X	X	X	X					X	X	32
	OPTIONS	asr	/CHANGE			X	X	X	X					X	X	31
	TYPE					X	X	X	X					X	X	16, 17
	All Others					X	X	X	X					X	X	
NAME						X	X	X	X					X	X	18
POOL						X	X	X	X					X	X	19
RTCODE					X	X	X	X	X					X	X	
SECURITY	PASSWORD	yes/no	/NRESTART											X	X	

Table 12. Macro Table - Selecting the Appropriate IMS System Definition (continued)

Macro	Operand	Value	JCL or Command	/DISPLAY	MODBLKS	CTLBLKS	NUCLEUS	ON-LINE	ALL	MSVERIFY	DB BATCH	TM BATCH	DBCTL	DB/DC	DCCTL	Refer to Table Notes at the end of this table
	TYPE	AGNEXIT					X	X	X				X	X	X	
		RACFAGN	ISIS=				X	X	X				X	X	X	2
		RACFCOM	RCF=				X	X	X					X	X	2
		RACFTERM	RCF=				X	X	X					X	X	2
		SIGNEXIT				X	X	X	X					X	X	23
		TRANEXIT				X	X	X	X					X	X	23
	SECCNT					X	X	X	X					X	X	
	SECLVL	SIGNON	/NRESTART			X	X	X	X					X	X	2, 22
		TRANAUTH	/NRESTART			X	X	X	X					X	X	2, 22
	TERMNL	yes/no	/NRESTART			X	X	X	X					X	X	
	TRANCMD	yes/no	/RESTART			X	X	X	X					X	X	
	All Others					X	X	X	X				X	X	X	
STATION						X	X	X	X					X	X	
SUBPOOL						X	X	X	X					X	X	
TERMINAL	ADDR		DD			X	X	X	X					X	X	15, 20
	KEY		/CHANGE			X	X	X	X					X	X	
	MODETBL		/CHANGE	nodename NAME		X	X	X	X					X	X	33
	OPTIONS	asr	/CHANGE			X	X	X	X					X	X	34
	XKEY		/CHANGE			X	X	X	X					X	X	
	All Others					X	X	X	X					X	X	
TRANSACT	DCLWA				X	X	X	X	X					X	X	
	EDIT	name				X	X	X						X	X	24, 25
		All Others			X	X	X	X	X					X	X	
	MAXRGN		/CHANGE		X	X	X	X	X					X	X	26
	MSGTYPE	class	/ASSIGN		X	X	X	X	X					X	X	1, 26

Table 12. Macro Table - Selecting the Appropriate IMS System Definition (continued)

Macro	Operand	Value	JCL or Command	/DISPLAY	MODBLKS	CTLBLKS	NUCLEUS	ON-LINE	ALL	MSVERIFY	DB BATCH	TM BATCH	DBCTL	DB/DC	DCCTL	Refer to Table Notes at the end of this table
		All Others			X	X	X	X	X					X	X	
	PARLIM		/ASSIGN		X	X	X	X	X					X	X	1, 26
	PROCLIM	count	/ASSIGN		X	X	X	X	X					X	X	1, 26
		All Others			X	X	X	X	X					X	X	
	PRTY		/ASSIGN		X	X	X	X	X					X	X	1, 26
	SEGNO		/ASSIGN		X	X	X	X	X					X	X	1, 26
	SEGSIZE		/ASSIGN		X	X	X	X	X					X	X	1, 26
	SYSID				X	X	X	X	X					X	X	1
	All Others				X	X	X	X	X					X	X	
TYPE						X	X	X	X					X	X	4, 15
VTAMPOOL						X	X	X	X					X	X	

Selecting Appropriate System Definition

Notes to Table 12 on page 32:

1. Changes to existing transactions cannot be introduced via a MODBLKS system definition if those attributes are changeable by online commands.
2. Refer to IMS, DBBATCH, DBC, DCC, and DLIBATCH procedures in “Chapter 4. Procedures” on page 189.
3. This keyword can be changed with any type of system definition except MODBLKS.
4. Adding or deleting VTAM support requires the minimum of an ON-LINE system definition.
5. Adding a user-supplied exit routine, user message table, or the shared printer message router exit routine requires the minimum of a CTLBLKS system definition. The exceptions to this requirement are the ETO, message greeting, and command security exit routines. To remove the use of these exit routines requires a minimum of NUCLEUS system definition. These exit routines are then loaded or deleted during initialization if they are in IMSRESLIB.
6. The ACCESS attribute is used in data sharing and can be dynamically changed with the /START command.
7. ACCESS can be forced to exclusive by specifying DISP=OLD in the database DD statement or in the DFSMDA definition.
8. Adding or deleting Fast Path support for the FPCTRL statement requires a minimum of an ON-LINE system definition. The FPBUF parameter of the TERMINAL macro is not required to enable Fast Path for a specific terminal. Fast Path is automatically enabled for all Fast Path-capable terminals.
9. Although the value for the Type 2 SVC number is changed in the necessary control blocks for all system definition types, only the ON-LINE and ALL types of system definition link the SVC module into IMS.RESLIB with the proper name required for subsequent link into the operating system libraries.
10. These parameters can only be **excluded** using JCL. If their inclusion in your system is required, they must be included using system definition parameters.
11. For an online system, modifying the IRLM= or IRLNM= parameters requires a minimum of a CTLBLKS type system definition. A batch system requires a minimum of either a BATCH or an ALL system definition.
12. Generating a system for use in an IMS backup configuration requires a minimum of an ON-LINE system definition.
13. Adding Database Surveyor support to an IMS system initially generated without it can be done in two ways. If done via system definition, either a BATCH generation (for batch) or an ALL generation (for both batch and online) must be specified to include Surveyor. Alternatively, in either a batch or online system, the following JCL can be run:

```
//LINKSUR JOB
//          EXEC PGM=IEWL,REGION=128K,
//          PARM='NCAL,LET,XREF'
//SYSPRINT DD SYSOUT=A
//LOAD     DD DSNAME=IMS.LOAD,DISP=SHR
//SYSLOAD  DD DSNAME=IMS.RESLIB,DISP=SHR
//SYSUT1   DD SPACE=(CYL,(1,1)),UNIT=SYSDA
//SYSLIN   DD *
           INCLUDE LOAD(DFSPRSUR)
           INCLUDE LOAD(DFSLI000)
           NAME DFSPRSUR(R)          DATABASE SURVEYOR
/*
```

14. Change the DD statement.
15. Adding or deleting support for a specific non-ETO terminal type requires a minimum of a NUCLEUS system definition. A minimum of a NUCLEUS system

Selecting Appropriate System Definition

definition is required if the SIZE= or FEAT= parameter is specified for the first time in an existing system or if the values of these parameters are changed. You can use any type of system definition except MODBLKS to add, delete, or change this macro statement.

16. Adding MSC support for the first time or deleting MSC support requires a minimum of an ON-LINE system definition. A physical link change requires a DD statement change.
17. Adding or deleting the BSC or channel-to-channel support requires a minimum of an ON-LINE system definition. Adding MSC VTAM or MTM support requires a minimum of an ON-LINE system definition. Deleting MSC VTAM or MTM support requires a minimum of a NUCLEUS system definition.
18. Removing remote LTERM support from an MSC system requires a minimum of a NUCLEUS system definition.
19. Adding or deleting switched terminal support requires a minimum of a NUCLEUS system definition.
20. The address of 3270 local terminals can be changed on the DD statements. All other terminals require a minimum of a CTLBLKS system definition.
21. When changing this parameter to FORCE, or from FORCE to another option, either a BATCH or an ALL system definition is required for the change to take effect in a batch environment.
22. JCL can be used to override SIGNON and TRANAUTH only if SIGNON is required or TRANSACT authorization is specified. The signon and transaction authorization exit routines can still be used if you sign on.
23. To remove exit routine support requires a minimum of a NUCLEUS system definition defined for static terminals.
24. MODBLKS generation cannot be used to add a transaction that is also adding a new user edit routine. MODBLKS generation cannot be used to change the order of the transactions if user edit routines are specified. The minimum generation for either of these occurrences is a CTLBLKS generation.
25. MODBLKS generation cannot be used to delete a transaction that has a user edit routine if this is the first or only transaction that is using this edit routine. MODBLKS generation cannot be used to change the order of the transactions if user edit routines are specified. The minimum generation for either of these occurrences is a CTLBLKS generation.
26. The command is /DISPLAY TRANSACTION.
27. The command is /DISPLAY POOL xxxx, where xxxx is the indicated parameter.
28. The command is /DISPLAY DATABASE.
29. The command is /DISPLAY ACTIVE.
30. The command is /DISPLAY LINE.
31. The command is /DISPLAY LINK.
32. The command is /DISPLAY LINK xxxx, where xxxx is the indicated parameter.
33. The command is /DISPLAY NODE.
34. The command is /DISPLAY MODE xxxx, where xxxx is the indicated parameter.
35. If Fast Path is to be included in a DCCTL environment, specify the FPCTRL macro without any parameters.
36. For batch systems, specify the parameter in the JCL; for online systems, specify the parameter in the DFSRSRxx member.
37. A BATCH or ALL system definition is required for the change to take effect in a batch environment.

Managing System Definition

System definition is a two-stage process. Stage 1 checks your input specifications and generates a series of MVS job steps for stage 2. Stage 2 builds IMS system libraries, execution procedures, and the IMS control program.

In an XRF environment, the system definition of the active and alternate IMS subsystems must be identical, although two separate system generations can occur.

Related Reading: See “Designing a System with Extended Recovery Facility” in *IMS/ESA Administration Guide: System* for information on defining an XRF system.

Sequencing Stage 1 Input

During the initialization for startup of the IMS online system, a control program, which has been customized to your requirements, is loaded. The input data that drives this process is a set of macro source statements. Each statement is coded with its own parameters. The composite of all the macro statements is termed the *stage 1 input*.

Six sets or groupings of macro statements make up the content of the stage 1 input. (The IMSGEN macro contributes to the system configuration definition.) Within each set, individual macros specify data that is specific to a required function or to a part of the total physical online configuration.

You can think of these groupings of macros as a type of hierarchical structure, as shown in Figure 3 on page 41. The group of required system configuration macros is shown as a root segment.

LGEN System Definition Changes

Two steps are required to define a large system generation:

1. In the IMSCTRL macro statement, specify LGEN as the fourth subparameter of the SYSTEM keyword. Define all other macro statements for the system generation as you normally do.
2. Allocate and catalog two additional data sets needed for the Stage 2 processing in the LGEN configuration. The default names for these data sets are IMS.LGENIN and IMS.LGENOUT. You can use the NODE keyword of the IMSGEN macro to override the default names. To allow sufficient storage, allocate ten primary and five secondary cylinders for each data set, and specify a PDS size of 40 blocks.

LGEN Stage 1 Processing

Stage 1 processing for LGEN system generation differs from a standard system generation. You must execute the preprocessor to assemble an LGEN generation. A stand-alone assembly of an LGEN generation causes an error. Stage 1 source statements are used as input and can include copy statements. However, the stage 1 input source statements cannot include inline copies of IMS macros, inline user macros, or user macro calls.

In turn, the preprocessor checks the resource name. The preprocessor uses storage above the 16 MB line and calls the assembler multiple times, called “cycles”, to assemble all the input. This ensures that the stage 1 execution requires only 4 MB of private storage. You receive two reports from stage 1 processing: a summary of the return codes for each assembly and a summary of error messages.

The result of the assembly cycles is:

- A job stream that creates members of a PDS for the IMS control blocks
- The job steps that are dictated by the system generation type

The PDS members are held in the LGENIN data set and are used as input to the Sort/Split utility. The Sort/Split utility sorts all the PDS members for a given resource type and then splits them into parts that are to be assembled in a 4 MB region. Each part becomes a member of a PDS stored in the LGENOUT data set. The Sort/Split utility supplies a report about its processing.

The Sort/Split utility processes each resource type individually, including both standard and customized resources. Customized resources, referred to as “user-defined resources”, are normally created by modifying stage 1 macros. A control record is required for each user-defined resource in the Resource Information File member. This record tells the Sort/Split utility to process the resource.

Related Reading: For more information on user-defined resources, see *IMS/ESA Customization Guide*.

LGEN Stage 2 Processing

LGEN stage 1 processing creates altered job steps for stage 2. However, the differences in stage 2 processing (between LGEN and standard) are not obvious because LGEN process stage 2 inputs into assemble and link edit jobs that are similar to the standard stage 2 job steps.

Hierarchy of Stage 1 System Definition Macros

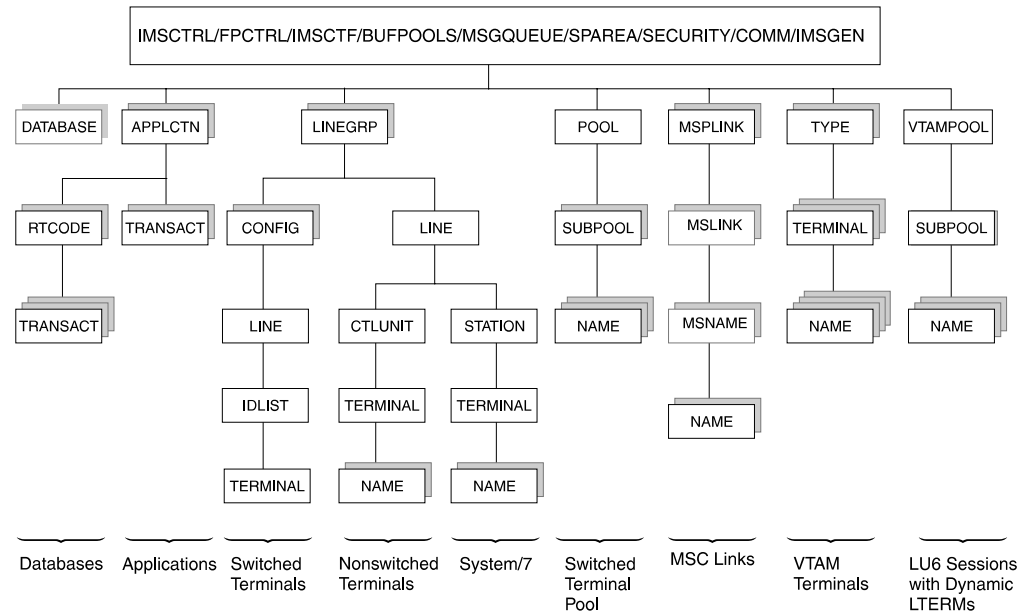


Figure 3. Hierarchy of Stage 1 System Definition Macros

Verifying System Definition

To verify system definition, you can use the system definition preprocessor to scan the stage 1 input. The preprocessor checks for duplicate names among the names you have defined and ensures that they are of the correct length and format. Assigned names are checked across resource types, too, so that transaction codes,

Verifying System Definition

LTERMs , and IMS subsystem names (used for multiple systems coupling) do not contain duplications. The preprocessor helps maintain the integrity of the stage 1 input stream.

Preprocessor

The system definition preprocessor is optional unless the LGEN configuration has been specified in the IMSCTRL SYSTEM keyword. Use the LGEN configuration when more than 4 MB of private storage are needed in stage 1 or in any stage 2 assembly job step.

If the IMSCTRL NAMECHK=NO option is selected for IMS system definition, run the system definition preprocessor.

In the LGEN configuration, the IMSCTRL NAMECHK=NO option is ignored and need not be specified. Under LGEN, no sorting is done in the stage 1 or stage 2 assemblies. Sorting is done in the preprocessor and by the Sort/Split utility.

Stage 1 source statements are used as input and can include copy statements. However, they cannot include inline copies of IMS macros from the stage 1 input, inline user macros, user macro calls, or conditional assembler statements.

When running LGEN, do not include assembler listing control instructions (such as PRINT OFF) in the stage 1 source statements. If you do so, you may have unpredictable results, because LGEN processes the output from the assembler listing.

The preprocessor locates, reads, and processes copy statements by using the copy members as input. The order of search, which conforms to MVS standards, depends upon the order in which you have concatenated the libraries. Copy members are retrieved from the data sets specified by SYSLIB DD statements. The preprocessor scans the input for the following IMS macros:

APPLCTN	NAME
DATABASE	RTCODE
MSLINK	SUBPOOL
MSNAME	TERMINAL
MSPLINK	TRANSACTION

The preprocessor does not recognize keywords that are split across continuation statements. The related parameters associated with these keywords generate error messages from the preprocessor.

After scanning for macros, the preprocessor builds resource name tables for each resource name type. It then performs the following verification steps:

- Within each resource type, the preprocessor verifies that no resources of that type have duplicate names. The following types of resource names are verified:
 - DBD names
 - PSB names
 - VTAM node names
 - MS link names
 - Logical terminal names
 - Transaction codes
 - Routing codes

Subpool names
MSLINK partner IDs
MSPLINK physical link names
Remote system VTAM node names

- These resource names are checked to ensure that they are of the appropriate length and are alphanumeric.
- The names specified for transactions, logical terminals, and multiple systems are cross-checked to ensure that no duplicate names exist across these three resource types.

When the preprocessor executes with LGEN specified, the following additional processing occurs:

- Stage 1 source statements are saved in storage above the 16 MB line according to resource type.
- The specifications for the system identification (SYSID) keyword on APPLCTN, TRANSACT, and MSNAME stage 1 source statements are checked for syntax errors. They are also cross-checked to ensure that no conflicts exist between the **local** system identification keywords and the **remote** system identification keywords.
- If no errors are detected by the preprocessor, the assembler is called multiple times to process the stage 1 source statements that were saved. Input to each assembly consists of all system macro statements and up to 10000 of either DB, MSC, or VTAM resources. All BTAM resources are processed during a single execution of the assembler.

The output of the preprocessor is a listing of the input records, a listing of diagnostic and error messages, and a return code.

If an LGEN configuration is active, the output includes the following:

- Two summary reports that are printed before any other preprocessor output. The first report contains the return code for each assembly cycle and the total number of resources processed by the preprocessor. The second report lists each error message in an assembly cycle and the five records that preceded the error message.
- Information from the listing of each assembly is appended to the preprocessor listing to form a consolidated listing data set. A list of the cycle macros is included when any assembly fails with a return code greater than 4. This list aids problem determination.
- The object data set from each assembly is appended to the object data sets of all previous assemblies to form a consolidated object data set. Execution of the job stream in this data set is functionally equivalent to stage 2 processing of a standard system definition for fewer resources.

You can optionally write two exit routines, DFSPRE60 and DFSPRE70, to perform additional processing.

Related Reading: Information about the register and coding requirements for the DFSPRE60 and DFSPRE70 exit routines can be found in *IMS/ESA Customization Guide*.

Executing the Preprocessor

To execute the preprocessor, you must provide JCL as follows:

Verifying System Definition

```
//          JOB
//          EXEC PGM=DFSPRE00,REGION=32M,PARM='xxx' 1
//STEPLIB DD DSN=IMS.RESLIB,DISP=SHR 2
//          DD DSN=USER.EXITLIB,DISP=SHR 3
//SYSLIB  DD DSN=USER.MACLIB,DISP=SHR 4
//          DD DSN=IMS.GENLIB,DISP=SHR 4
//          DD DSN=IMS.GENLIBA,DISP=SHR 5
//SYSIN80 DD DSN=&&SYS80, 5
// UNIT=SYSDA,SPACE=(CYL,(5,5)), 5
// DCB=(BLKSIZE=11440,LRECL=80,RECFM=FB,DSORG=PS) 5
//SYSUT1  DD DSN=&SYSUT1, 5
// UNIT=SYSDA,SPACE=(CYL,(10,5)) 5
//SYSLIN  DD DSN=&&SYSLN, 5
// UNIT=SYSDA,SPACE=(CYL,(1,1)), 5
// DCB=(BLKSIZE=11440,LRECL=80,RECFM=FB,DSORG=PS) 5
//SYSPRT80 DD DSN=&&SYSPRT, 5
// UNIT=SYSDA,SPACE=(CYL,(5,5)), 5
// DCB=(BLKSIZE=3146,LRECL=121,RECFM=FBM,DSORG=PS) 5
//SYSSUMPR DD SYSOUT=A 6
//SYSCYLPR DD SYSOUT=A 6
//SYSPRINT DD SYSOUT=A
//SYSCOBJ DD DSN=IMS.SYSCOBJ,DISP=(NEW,KEEP), 7
// UNIT=SYSDA,SPACE=(CYL,(5,5)), 7
// DCB=(BLKSIZE=11440,LRECL=80,RECFM=FB,DSORG=PS) 7
//SYSIN   DD * 8
```

Notes:

1. The region size indicated is the size required to execute the preprocessor on an MVS/ESA system. The LGEN subparameter is specified in the SYSTEM= keyword on the IMSCTRL source statement. Guidelines for determining the region sizes without LGEN are in "Estimating Storage Requirements for the Preprocessor".
The PARM field is specified only if the default exit routine indicators should be overridden. See the following section, "Defining Preprocessor Parameters", for information on use of the PARM field.
2. This DD statement should specify the library containing the preprocessor version that you want to execute. Different versions can exist because of SMP maintenance.
3. This concatenated DD statement is required only if exit routines are requested.
4. The SYSLIB DD statement should point to a library that contains the copy members, but with an LGEN definition, the statement should point to desired system definition macro instruction libraries. This DD statement is required if stage 1 source COPY statements or an LGEN configuration exists.
5. Only the LGEN configuration requires these DD statements.
6. Only the LGEN configuration requires these DD statements. These DD statements must appear before the SYSPRINT DD statement to allow the summary reports to precede the consolidated assembly listing.
7. Only the LGEN configuration requires this data set; it contains the consolidated object module from all the LGEN assembler cycles. The input/output unit assigned to this data set can be a card punch or an intermediate storage device capable of sequential access.
8. You can also use the statement:

```
//SYSIN DD DSN=....,DISP=SHR
```

Defining Preprocessor Parameters

The PARM field on the EXEC statement is specified as follows:

```
PARM='a,b,(asmopt1,asmopt2,...,asmoptn)'
```

Verifying System Definition

The parameters specified are positional and are specified as follows:

- a** Is a value of 'Y' or 'N' and indicates whether (Y) or not (N) exit routine DFSPRE60 is to be used during this invocation of the preprocessor. If DFSPRE60 is to be used, it must reside on the libraries pointed to by the STEPLIB DD statement. The default is N.
- b** Is a value of 'Y' or 'N' and indicates whether (Y) or not (N) exit routine DFSPRE70 is to be used during this invocation of the preprocessor. If DFSPRE70 is to be used, it must reside on the libraries pointed to by the STEPLIB DD statement. The default is N.

asmopt1, ..., asmoptn

Are any assembler options you want for stage 1 assemblies in the LGEN configuration. Enclose the options in parentheses, and use the comma as a delimiter when using more than one option. The preprocessor forces the LIST and OBJECT options.

Related Reading: For more information on the LGEN configuration, refer to *IMS/ESA Administration Guide: System*.

Estimating Storage Requirements for the Preprocessor

For both the standard and LGEN system generations, the preprocessor uses the default values to reserve an initial amount of storage for each resource name table (RNT). If this storage is insufficient, the preprocessor dynamically expands the table until all extended private storage is exhausted. Resource names not added to the appropriate RNT are ignored and excluded from further processing.

Additional space is required if the number of errors for any given resource name exceeds 50, or if the number of nested copy statements exceeds 50. For most cases, this additional storage is accounted for in the base system storage number.

The storage size is the minimum region size that can be specified on preprocessor invocation JCL.

The preprocessor also requires one megabyte of private storage below the 16 MB line.

Standard System Generation

The storage estimate is for extended private storage. The storage requirement is the sum of:

1. 1000 KB for the base system
2. Space needed for exit routines DFSPRE60 and DFSPRE70, if they are loaded
3. Table space in excess of the default. For each of the following resource types that exceeds 5120, determine the amount of storage from Table 13 on page 46.

- APPLCTN
- DATABASE
- NAME
- RTCODE
- SUBPOOL
- TERMINAL (VTAM)
- TRANSACT

Verifying System Definition

Table 13. Resource Name Table Storage

Resource Range		Storage Amount
from	to	in K bytes
5120	10240	80
10240	20480	240
20480	41060	560
41060	82120	1200
82120	164240	2480
164240	328480	5040
328480	656960	10160
656960	1313920	20400

LGEN System Generation

If the total number of resources is less than 200000, allow 32 MB for extended private storage.

When the total number of resources exceeds 200000 resources, determine the amount of extended private storage that is required.

The sum of the following items, numbered 1 and 2, is the total extended private storage (in kilobytes) required to run the preprocessor with an LGEN system generation. The preprocessor also requires 4 MB of private storage below the 16-MB line.

LGEN limits the amount of private storage required to 4 MB for DB, MSC, and VTAM resources. All BTAM resources are assembled in a single assembly cycle. A large number of BTAM resources can require more than 4 MB of private storage for the stage 1 and stage 2 assemblies.

- Storage for system generation input is the sum of:
 - 3000 for the base system
 - ((Number of system generation input records/8192)x 584)
 - 584 if any BTAM resources are defined
 - 584 if any MSC resources are defined
 - 584 if any VTAM resources are defined
- Table space in excess of default. For each of the following resource types that exceeds 5120, determine the amount of storage from Table 13.
 - APPLCTN (total)
 - APPLCTN (remote)
 - DATABASE
 - NAME
 - RTCODE
 - SUBPOOL
 - TERMINAL (VTAM)
 - TRANSACT (total)
 - TRANSACT (remote)

Running System Definition Stage 1

After coding your macro statement input (and optionally running the preprocessor), you must now run your IMS system definition stage 1. The preprocessor runs stage 1 in an LGEN configuration. The stage 1 output is the input to system definition stage 2. The two stages of the system definition process are illustrated in Figure 4.

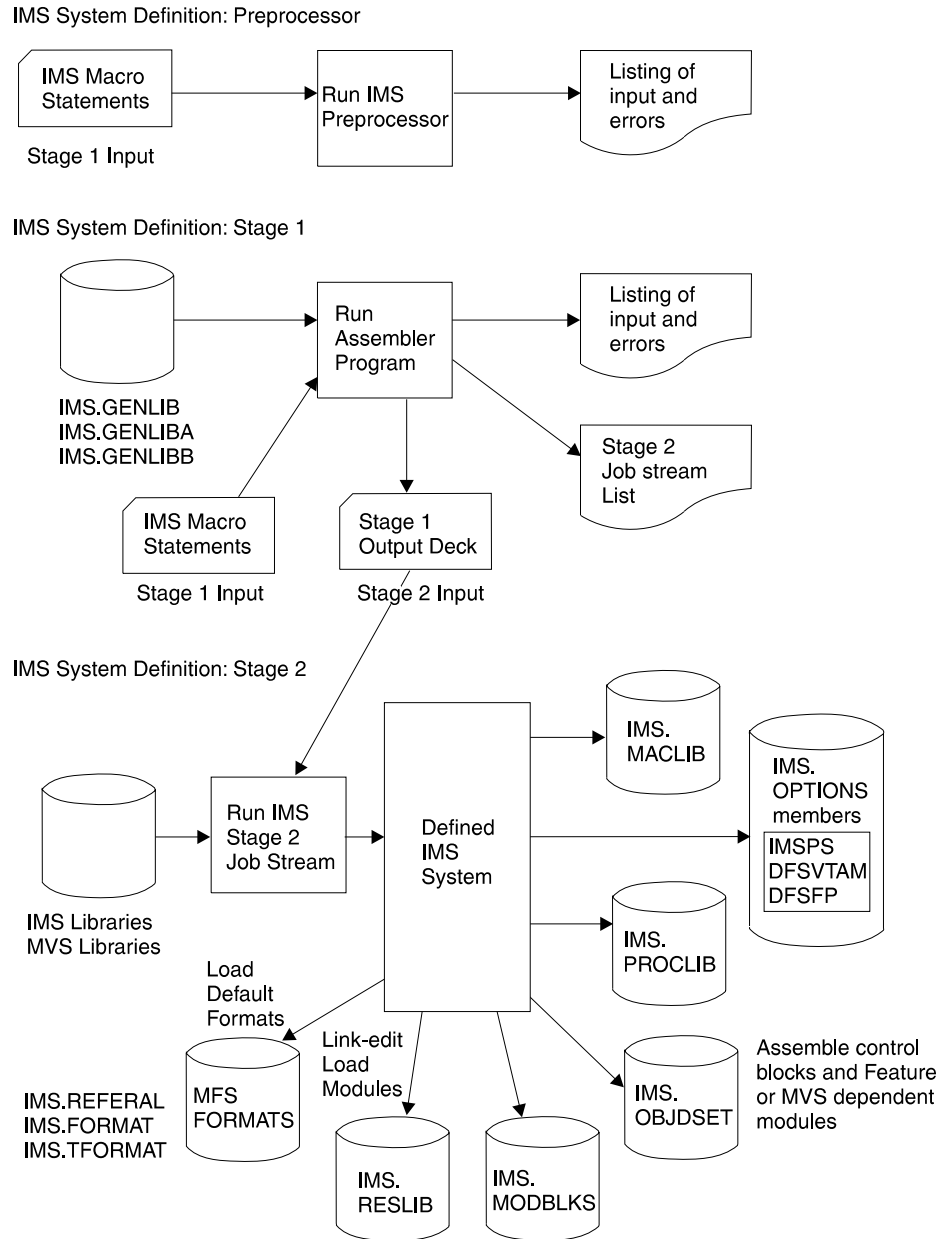


Figure 4. Summary of the Two Stages of System Definition Processing

Entering Stage 1 Input Statements

Stage 1 of IMS system definition consists of running the assembler with the IMS macro statements that you have prepared as input. The macro sets must be placed

Running Stage 1

in the order described in “Sequencing Stage 1 Input” on page 40. If more than 5000 objects are defined, sufficient storage must be allocated for the system definition process.

Stage 1 and Extended Terminal Option

If you are using the Extended Terminal Option (ETO) of IMS, you can run stage 1 in three different ways.

- You can use the existing stage 1. In this case, the default ETOFEAT is NO on the IMSCTRL macro, and the output from stage 1 is the same as described in Figure 4 on page 47.
- You can run the existing stage 1 with one modification. By coding ETOFEAT=(YES,,ONLY) on the IMSCTRL macro, the output from stage 1 is the only stage 2 step required to create the ETO descriptor member and the ETO descriptor report. Use this method only for standard system generation.
- You can code ETOFEAT=YES to generate an ETO system if the ETO feature is installed.

For more information on ETO, see “Enabling ETO Support for ACF/VTAM Terminals” on page 399.

Stage 1 Output

Output that results from the successful execution of the stage 1 input includes:

- The MVS job stream for stage 2.
- An output listing that also includes a listing of the input statements you submitted and comments about the system you defined.

Printed Output

The printed stage 1 output lists the input statements that you prepared, and the output. Input errors or inconsistencies are noted on the listing. Included are explanations of tasks that must be accomplished before stage 2 of IMS system definition can be executed.

Related Reading: If errors are encountered during IMS system definition stage 1, error messages that appear on the output listing are explained in *IMS/ESA Messages and Codes* or in the appropriate assembler language messages manual.

One coding error that frequently occurs is beginning a continued statement in column 17 or later. The assembler does not flag this as an error. Rather, the improperly continued statement is treated as a comment. IMS therefore assumes the appropriate default values for any operands that were coded on the continuation statement and ignores any values you might have specified.

Link-Edit Order Statements

The ORDER statements produced by stage 1 must be used when link-editing the IMS nucleus.

Stage 2 Input

The stage 1 output is the input to IMS system definition stage 2. If possible, you should list these statements again for easier use and become familiar with them before performing stage 2.

Some or all of the job steps are described below, depending on the type of IMS system definition specified in the IMSCTRL macro statement.

Running Stage 1

- Creation of IMS.MACLIB by copying either or both of IMS.GENLIBA or IMS.GENLIBB.
- Creation of IMS.OPTIONS containing IMSPS, DFSVTAM, DFSFP, and DFSIRLM.
 - The IMSPS member is used during the APAR maintenance process.
 - The DFSVTAM member is used in the stage 2 assembly of the sysgen-created module, DFSCLOx, where x is the sysgen suffix. It provides information as to whether VTAM or BTAM terminals were defined in stage 1.
 - The DFSFP member is used by conditional assemblies during stage 2 processing. It provides information as to whether Fast Path was defined in stage 1.
 - The DFSIRLM member is used by conditional assemblies during stage 2 processing. It provides information as to whether the IRLM component is defined in stage 1.
- Creation of IMS.PROCLIB.
- Assemblies of online control blocks into IMS.OBJDSET.

In some cases, the region size generated for the jobs that assemble the IMS control blocks cannot be adequate.

If a region larger than 4 MB seems necessary, running the stage 1 with an LGEN configuration ensures that stage 2 job steps require no more than 4 MB.
- Link-edits of reentrant, reusable, and default attribute load modules and control block modules into IMS.RESLIB and IMS.MODBLKS.
- Program executions to load the Message Format Service (MFS) default formats into IMS.REFERAL, IMS.FORMAT, and IMS.TFORMAT.

The complete output from stage 1 must be entered into the MVS/ESA job stream in the order that it was produced. This initiates stage 2 of your IMS system definition.

Running System Definition Stage 2

You must put the stage 1 output job stream in your MVS system's input job stream to initiate stage 2 of the IMS system definition. If stage 2 processing is to be multiprogrammed, the job step that creates IMS.OPTIONS must complete before any assemblies are run. Assemblies must complete before link-edits are performed, and the nucleus link-edit job must not run until other link-edits are complete. If stage 2 processing is to be multiprogrammed, then the assemblies must be run **sequentially** rather than in parallel. Successful execution of stage 2 of system definition completes this IMS installation step. All resources, except communication lines and message processing regions, are generated in a "started" mode; a /START command is not required to make the resource operational.

Stage 2 Output

The JCL from stage 1, which is used as input to stage 2, shows that SYS1.MACLIB and SYS1.MODGEN are concatenated to any or all of IMS.OPTIONS, IMS.GENLIB, IMS.GENLIBA, or IMS.GENLIBB. As in all concatenated libraries, the block sizes of the first library must be at least as large as the largest block size specified for the libraries concatenated to it.

The assembly and link-edit listings produced during stage 2 execution indicate whether execution was successful. All of these job steps should complete with a 0 return code.

If condition code 4 is set in the job that creates MFS default formats, it should be ignored.

Maintaining Stage 1 Output with SMP/E

To ensure that your IMS system is installable and maintainable via SMP/E, you must run the SMP/E JCLIN against your stage 1 SYSPUNCH output. This informs SMP/E of the contents of the IMS system libraries. SMP/E JCLIN must be run against the stage 1 SYSPUNCH output of all types of system definition to ensure that your SMP/E control data set remains current; that is, the SMP/E data set must reflect the most current status of resources defined in your online system.

Large System Definition SMP/E JCLIN

LGEN imposes some changes in the way SMP/E JCLIN is performed:

- JCLIN must be run either after Stage 2 or after the LGEN utility steps (at the beginning of the Stage 2) are run.
- The LGEN Stage 2 must be converted to a non-LGEN format before JCLIN is run. Program DFSIVG10 is provided for this purpose.

Sample JCLIN Job

The following is an example of the JCLIN job that you should run. This is an example using SMP/E. If you are using SMP/4, you must also change the procedure name for the JCLIN processing step. Use the job stream built from this example for both LGEN and non-LGEN jobs.

```
//      JOB
//*
//*****
//*NOTE:  FOR NON-LGEN SYSTEM DEFINITIONS, SMP JCLIN
//*  PROCESSING MAY BE RUN EITHER BEFORE OR AFTER STAGE2.
//*
//*NOTE:  FOR LGEN SYSTEM DEFINITIONS, SMP JCLIN PROCESSING
//*  MAY ONLY BE RUN EITHER:
//*
//*  1 - AFTER STAGE2 PROCESSING IS COMPLETE.
//*  2 - AFTER THE LGEN STEPS AT THE BEGINNING OF THE
//*      STAGE2 JOB STREAM HAVE BEEN RUN
//*****
//*
//*SCRATCH THE TEMPORARY DATA SET USED BY THE STAGE2 POST
//*PROCESSOR PROGRAM.
//*
//SCRATCH EXEC PGM=IEHPROGM
//SYSPRINT DD SYSOUT=*
//IMS  DD DISP=OLD,UNIT=SYSDA,VOL=SER=IMS31
//SYSIN  DD *
//      UNCATLG DSNAME=IMS.STG2OUT
//      SCRATCH DSNAME=IMS.STG2OUT,
//              VOL=3380=IMS31
//*
//*ALLOCATE THE TEMPORARY DATA SET USED BY THE STAGE2
// POST PROCESSOR PROGRAM.
//*
//ALLOCATE EXEC PGM=IEFBRI4
//STG2OUT DD DSN=IMS.STG2OUT,
//      DISP=(NEW,CATLG),
//      UNIT=SYSDA,
//      VOL=SER=IMS31,
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160),
//      SPACE=(CYL,(1,5))
//*
```



```
//*POST PROCESS THE STAGE2 JOB STREAM PRIOR TO SMP JCLIN.
//*
**FOR NON-LGEN SYSTEM DEFINITIONS: THE STAGE2 JOB STREAM
**IS COPIED UNCHANGED TO THE TEMPORARY DATA SET.
**
**FOR LGEN SYSTEM DEFINITIONS: AS THE STAGE2 JOB STREAM
**IS COPIED TO THE TEMPORARY DATA SET, THE MEMBERS OF THE
**LGENOUT DATA SET ARE PLACED IN-LINE IN THE STAGE2 JOB
**STREAM PRODUCING A STAGE2 JOB STREAM THAT LOOKS LIKE A
**NON-LGEN STAGE2.
** //      COND=(4,LT,POSTPROC)
//SMPCNTL DD *
  SET BDY(IMSTZ31).
  JCLIN ASM(PGM=IEV90) JCLINREPORT.
//SMPJCLIN DD DISP=OLD,
//      DSN=IMS.STG2OUT//POSTPROC EXEC PGM=DFSIVG10
//STEPLIB DD DISP=SHR,DSN=IMS.RESLIB
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//STG2IN DD DISP=SHR,
//      DSN=IMS.STG2IN
//STG2OUT DD DISP=OLD,
//      DSN=IMS.STG2OUT
//LGENPDS DD DISP=SHR,
//      DSN=IMS.LGENOUT
//*
**PERFORM SMP JCLIN PROCESSING
**
//JCLIN EXEC PROC=IMSSMPE,
```

The data set names, DASD volume name, and DASD volume device might need to be changed.

Chapter 3. Macros

Use macro statements, such as APPLCTN or TERMINAL, to generate a sequence of assembler statements from a single source statement. IMS provides a set of macro statements for system definition. For example, to describe program resource requirements for your application programs, you would use the APPLCTN macro. To define characteristics of VTAM nodes and non-VTAM communication, use the TERMINAL macro.

For a table of macros with the system definition types applicable to each macro, see Table 12 on page 32. For information on how to interpret syntax diagrams, see “Syntax Diagrams” on page xv in the preface section of this book.

In this Chapter:

- “Guidelines for IMS Macros.” on page 54
- “APPLCTN Macro” on page 56
- “BUFPOOLS Macro” on page 61
- “COMM Macro” on page 64
- “CONFIG Macro” on page 73
- “CTLUNIT Macro” on page 74
- “DATABASE Macro” on page 75
- “FPCTRL Macro” on page 76
- “IDLIST Macro” on page 78
- “IMSCTF Macro” on page 79
- “IMSCTRL Macro” on page 82
- “IMSGEN Macro” on page 95
- “LINE Macro” on page 108
- “LINEGRP Macro” on page 112
- “MSGQUEUE Macro” on page 116
- “MSLINK Macro” on page 120
- “MSNAME Macro” on page 122
- “MSPLINK Macro” on page 123
- “NAME Macro” on page 127
- “POOL Macro” on page 132
- “RTCODE Macro” on page 132
- “SECURITY Macro” on page 133
- “SPAREA Macro” on page 137
- “STATION Macro” on page 137
- “SUBPOOL Macro” on page 139
- “TERMINAL Macro” on page 140
- “TRANSACT Macro” on page 173
- “TYPE Macro” on page 186
- “VTAMPOOL Macro” on page 188

Guidelines for IMS Macros.

This chapter presents the IMS system definition macro statements in alphabetic order. Each macro statement description contains a figure showing the macro's syntax and an explanation of the macro's keywords and parameters.

"Resource Naming Rules" describes rules and restrictions that apply to IMS macros; "Coding Conventions" on page 56 describes coding conventions for macros and examples.

Resource Naming Rules

These rules and restrictions apply to all IMS macros.

- Names cannot include a blank, comma, period, hyphen, or equal sign.
- All PSB names must begin with an alphabetic character (alphabetic = A through Z, #, \$, and @) followed by zero to seven alphanumeric characters.
- Logical terminal names and transaction codes must be one to eight alphanumeric characters in length. (Alphanumeric = A through Z, #, \$, and @, 0 through 9.) Each name within the set comprising LTERM names and transaction codes must be unique.
- User IDs that are passed to RACF must be one-to eight-alphanumeric characters, that is, letters A through Z, digits 0 through 9, and the national characters #, \$, and @.

For ETO terminals, if the user ID is used to build the LTERM structure, the user ID must be one-to eight-alphanumeric characters.

- Node names must be unique among themselves, but can duplicate LTERM names, transaction codes, or subpool names.
- Subpool names must be unique among themselves, but can duplicate LTERM names, transaction codes, or node names.
- IMS null words **must not** be used as resource names (FOR, TO, ON, AFTER, and SECURITY). Also, resource names **must not** begin with DFS (except for DFSIVPxx and DFSSAMxx) or DBCDM; nor should WTOR, MSDB, SDB, DBRC, BASICEDT, or ISCEDT and its aliases be used as a resource name.
- Command keywords must not be used as resource names.

Related Reading: See *IMS/ESA Operator's Reference* for more information on command keyword restrictions.

- Each IMS macro statement can appear in an IMS system definition a limited number of times. Table 14 on page 55 shows the maximum number of times each macro statement can occur.
- Symbolic names can be used to define 3270 and SLU Type 2 devices. See 173 and the description of the to the TYPE= keyword of the TERMINAL macro statement for the format of these names. A symbolic name must be used to define the format of any 3270 display device with a screen size other than 480 or 1920 characters.
- The maximum number of IMS resources that can be defined is a design limit (that is, a theoretical limit). You should not assume that your installation can successfully define the maximum number of IMS resources (listed in Table 14 on page 55). The actual limit on this number is influenced by many limiting factors in your hardware and software configuration.

Resource Naming Rules

Table 14. Maximum Occurrences of Each IMS System Definition Macro Statement

Macro Statement	Maximum Occurrences/ Specification the assembler
APPLCTN	999,999
BUFPOOLS	1
COMM	1
CONFIG	255
CTLUNIT	1000
DATABASE ¹	32,700
FPCTRL	1
IDLIST	1000
IMSCTF	1
IMSCTRL	1
IMSGEN	1
LINE	1000 - (Number of POOL macros specified)
LINEGRP	676 - (Number of MSPLINK macros specified)
MSGQUEUE	1
MSLINK	675
MSNAME	676
MSPLINK	676 - (Number of LINEGRP macros specified)
NAME	999,999 - (Number of switched line macros specified)
POOL	1000 - (Number of LINE macros specified)
RTCODE	999,999
SECURITY	1
STATION	999,999 - (Number of TERMINAL, SUBPOOL, and CONFIG macros specified)
SUBPOOL	999,999 - (Number of TERMINAL, STATION, CONFIG macros specified)
TERMINAL	999,999 - (Number of SUBPOOL, CONFIG, and STATION macros specified)
TRANSACT	999,999
Transaction Edit Routines ²	255
TYPE	999,999
User routines ³	200
VTAMPOOL ⁴	1

Notes:

1. In IMS execution, database allocation is limited to the number of ddnames that MVS allows.
2. Transaction Edit routines are specified in the EDIT operand of the TRANSACT macro. Although the limit of TRANSACT statements is 999,999, only 255 transaction edit routines are accepted.
3. *User Routines* refer to the user-supplied physical terminal output and input edit routines (specified in the EDIT operand of the LINEGRP and TYPE macros).

Resource Naming Rules

4. VTAMPOOL is used once for each set of LU 6.1 subpools defined. It can be used multiple times within a system definition to define multiple sets of LU 6.1 subpools.

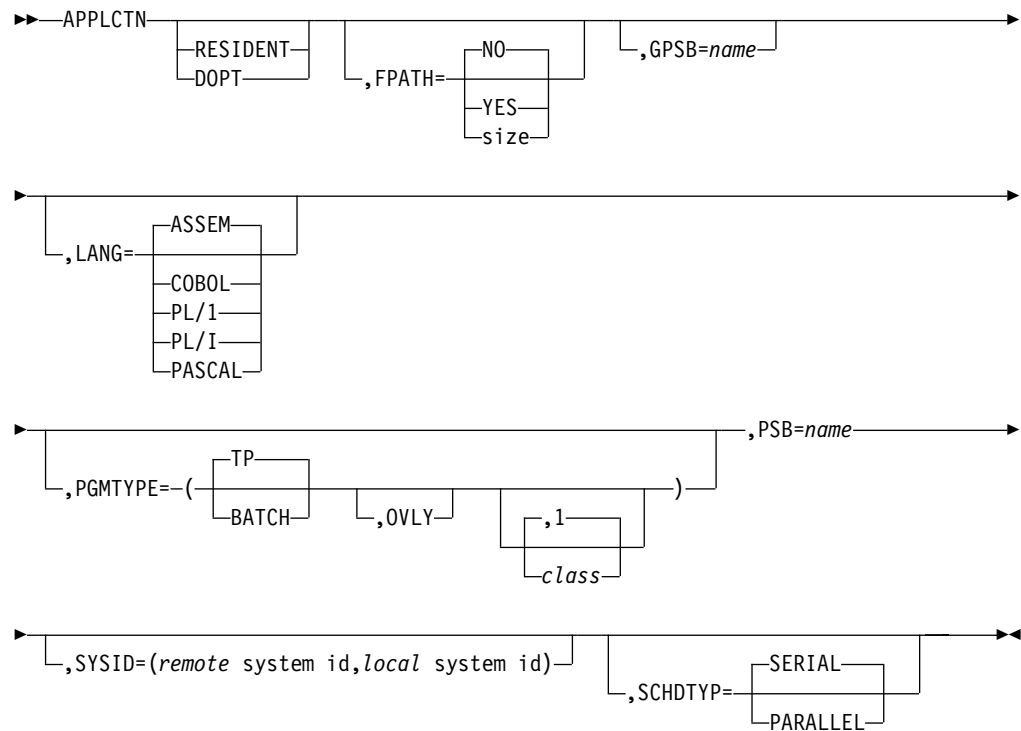
Coding Conventions

Both this book and its co-requisite, *IMS/ESA Installation Volume 1: Installation and Verification*, use the following conventions in coding macros and examples:

- Uppercase letters, stand-alone numbers, and punctuation marks must be coded exactly as shown. The only exceptions are subscripts, which are not coded.
- Lower-case letters, words, and associated numbers represent variables for which specific values can be substituted.
- A space or the lower-case “b” indicates one blank position.
- If an alternative item is underlined, it is the default. IMS assumes that if none of the items is coded, the underlined item is automatically the choice.
- You must specify positional parameters in the order shown.
- You can specify keyword parameters in any order.

APPLCTN Macro

The APPLCTN macro allows you to define the program resource requirements for application programs that run under the control of the IMS DB/DC environment, as well as for applications that access databases through DBCTL. An APPLCTN macro combined with one or more TRANSACT macros defines the scheduling and resource requirements for an application program. Using the APPLCTN macro, you only describe programs that operate in message processing regions, Fast Path message-driven program regions, batch message processing regions, or CCTL threads. You do use the APPLCTN macro to describe application programs that operate in batch processing regions. When defining an IMS data communication system, at least one APPLCTN macro is required.



Positional Parameters

The positional parameter RESIDENT specifies that the PSB associated with this application program is to be made resident during system initialization. RESIDENT and DOPT (dynamic PSB option) are mutually exclusive. DOPT and SCHDTYP=PARALLEL are also mutually exclusive.

DOPT specifies that the PSB associated with this application program is to be located dynamically. If DOPT is specified, the following actions are taken during the execution of the IMS control region:

- Initialization does not perform a BLDL on the PSB associated with this application program. Thus the PSB does not need to be in any data set defined by the ACBLIB DD statement until it is actually required to process a transaction. This is not true, however, of the DBDs that the PSB might reference. All DBDs that are to be used online must be available during initialization in a data set defined by the ACBLIB DD statement. PSBs referencing DBDs that are missing during initialization are not scheduled.
- Each time the program associated with this PSB is scheduled, a BLDL is performed, and the latest copy of the PSB is located. A BLDL is not performed for associated DBDs, so the DBD cannot be modified until the system is reinitialized or until online change is used.
- When the program terminates, the PSB is deleted from the PSB pool as part of the termination process.

Neither RESIDENT nor DOPT is the default parameter. Rather, if neither RESIDENT nor DOPT is selected on the APPLCTN macro statement, IMS system initialization causes a BLDL to be performed on the PSB associated with the application being defined. The PSB is not made resident (that is, loaded from the ACBLIB) until the application is scheduled.

APPLCTN Macro

To provide greater flexibility in the user's control of the library containing the dynamic PSBs, the following restrictions are imposed:

- The PSB must reside in a library other than the primary ACBLIB and must be concatenated to it.
- The concatenated library containing the PSB must be in ACBLIB format.
- The IMS.PSBLIB data set cannot be used.

If the BLDL performed at the scheduling of the PSB determines that the PSB resides in the first concatenation of the IMSACB DD statement set, the PSB is stopped and an error message is sent to the master terminal. The PSB is not scheduled. If the dynamic PSB is added to the concatenation of the IMS.ACBLIB, causing the data set to be expanded to a secondary extent, that PSB is not available to the online system until the ACBLIB is closed and reopened by IMS.

For MSC remote applications, the APPLCTN macro provides documentation and a reference for its transactions. It does not generate a control block. The PSB need not be generated in systems where it is used as a remote reference. The PSB name is defined here for documentation purposes.

To dynamically reassign a transaction from remote to local processing, a control block must be present in the local system.

Related Reading: See the description of the /MSASSIGN command in *IMS/ESA Operator's Reference* for additional information on dynamic reassignment.

Thus, if dynamic reassignment is desired, a transaction must be defined as local. That is, it must be under a local application having the same PSB name as the remote application (using the APPLCTN macro without a SYSID parameter). If dynamic reassignment is not desired, local definition is unnecessary.

Online changes to both the RESIDENT and DOPT options can be made via a MODBLKS system definition. However, PSBs associated with new application programs defined as RESIDENT (that is, changed from NONRESIDENT or DOPT to RESIDENT) during a MODBLKS system definition are not made resident until the next restart of IMS. Until that time, they are be treated as nonresident. Online changes made to already-resident PSBs (via an ACBGEN) cause the PSB to be treated as nonresident until the next IMS restart.

MPPs scheduled against dynamic PSBs are not allowed to go through the quick reschedule process or become pseudo WFIs.

Related Reading: For additional information on PSBs defined with DOPT see *IMS/ESA Utilities Reference: System* under "Application Control Blocks (ACB) Maintenance Utility."

Keyword Parameters

To find which parameters apply to your IMS configuration, refer to Table 12 on page 32.

FPATH=

Specifies whether (YES) or not (NO) this is a Fast Path-exclusive application program. FPATH=size, which determines the EMH buffer size required to run the transaction, overrides the EMHL execution parameter and implies FPATH=YES. The minimum specification for FPATH=size is 12; the maximum is 30720. FPATH=YES implicitly defines a wait-for-input (WFI) application

program. The PGMTYPE= parameters that define the overlay structure and class are invalid if FPATH=YES is specified. The SYSID= parameter is also invalid if FPATH is specified.

If FPATH=YES is specified during a MODBLKS system definition, Fast Path must have been previously defined for the online system to which this change is made.

Fast Path-potential transactions must be able to run under two applications. One of these applications must be defined with FPATH=YES; the other with FPATH=NO. The application with FPATH=YES must also be defined with either the same transaction or with a routing code that can be assigned by the user Input Edit/Routing exit routine.

Related Reading: See the TRANSACT macro in this book, and also the *IMS/ESA Customization Guide* for information about the user Input Edit/Routing exit routine.

When Fast Path is included during system definition, the FPBUF parameter on the TERMINAL macro statement is ignored, except to determine the default EMH buffer size. Fast Path buffers are provided by the EMHB pool, which expands and contracts dynamically depending on the number of ETO terminals concurrently entering Fast Path transactions.

GPSB=

Causes the scheduling process of all environments to generate a PSB containing an I/O PCB and an alternate modifiable PCB. With the GPSB= keyword, you do not need to perform the PSBGEN and ACBGEN, thus eliminating I/O to the ACBLIB.

GPSB= generates an I/O PCB named IOPCBbbb; The modifiable, alternate PCB is named TPPCB1bb. With an alternate modifiable PCB, an application can use the CHNG call to change the output destination and send output to a destination other than the input destination.

You can make an online change to add the GPSB option to an existing application, or add a new application with the GPSB options by using a MODBLKS system definition. However, the GPSB option does not take effect unless the ACBLIB is also changed with online change.

LANG=

Defines the language interface of the application program. You can use the following values with the LANG= keyword:

ASSEM is the default value.

If GPSB= is specified, LANG= is used to determine the language; it is used only with GPSB=.

PGMTYPE=

Specifies application program characteristics.

The first parameter describes the type of application program being defined. The default, TP, specifies that IMS schedules the program when messages processed by the program exist in the system. A program defined as BATCH can use DL/I in the IMS control program system region and can refer to the

APPLCTN Macro

message queues. If BATCH is coded, all TRANSACT macro statements that follow are assigned normal and limit priority values of zero.

If a program is changed from batch to online using the online change facility, you must enter the /ASSIGN command to assign nonzero current, limit, and normal priorities (CPRI, NPRI, and LPRI keywords) to the transactions using that program. This is because the online change facility does not alter attributes that are changeable through the /ASSIGN command. Regardless of whether you specify new values for the transaction in the MODBLKS system definition, they are ignored during /MODIFY processing.

If FPATH=YES is specified

- TP specifies a message-driven Fast Path application program.
- BATCH cannot be specified. If BATCH is specified, an error message is issued. Fast Path nonmessage-driven application programs are not supported, and should be changed to run as BMPs.
- The PGMTYPE= parameters that define the overlay structure and class are invalid.
- The SYSID= parameter is invalid.

The OVLY parameter on the APPLCTN macro statement is no longer used by the system but is retained for compatibility. An execution time parameter, OVLA on procedure DFSMPR, is available when starting an MPP to indicate whether the overlay supervisor should be preloaded by IMS. For information on coding DFSMPR, see "DFSMPR Procedure" on page 211.

The third parameter of the PGMTYPE= keyword specifies the class to which the transaction codes specified in the following TRANSACT macro statements are to be assigned. This parameter must be a decimal number from 1 to 255. This value must not exceed the value given (by specification or default) on the MAXCLAS= keyword of the IMSCTRL macro. The default is 1. If the transaction code class is to be specified in the individual TRANSACT macro statements, this parameter need not be coded. If the transaction code class is specified in both the APPLCTN and TRANSACT macro statements, the APPLCTN macro specification is ignored, and the TRANSACT macro specification is used.

If the PGMTYPE= (.,class) parameter is to be changed online, the class value that is specified cannot exceed the definition (by specification or default) on the MAXCLAS= keyword of the IMSCTRL macro in the online system to which this change is to be made.

The numeric class subparameter must not be specified if FPATH=YES is specified.

PSB=

Specifies the name of the PSB associated with this application program definition. Each local PSB name must be unique. A remote and a local application can each be defined as having the same PSB name. This is required in order to dynamically reassign a transaction from remote to local processing. The first character of the PSB name must be a letter. If PGMTYPE=TP, the PSB name must also be the program name.

Related Reading: For the description of the /MSASSIGN command, see *IMS/ESA Operator's Reference*.

SYSID=

Specifies in the multiple-IMS system configuration, the system identification (SYSID) of the remote system (that system on which the application executes) and the SYSID of the local system (the originating system to which the responses are returned). The values specified must be numbers in the range from 1 to 2036.

The remote SYSID specified must also be defined in an MSNAME macro statement, but the local SYSID can be defined in any or all of the MSNAME, TRANSACT, and APPLCTN macro statements. If SYSID is specified, all other keywords except PSB are ignored.

If the SYSID parameter is specified in the APPLCTN macro statement, you need not specify the SYSID in the TRANSACT macro statement. If the SYSID is specified in both the APPLCTN and TRANSACT macro statements, the APPLCTN specification is ignored.

The SYSID parameter is independent of the link type (BSC, CTC, MTM, VTAM) specified on the TYPE= keyword of the MSPLINK macro statement.

Because the values associated with the SYSID= keyword can be changed using an /MSASSIGN command, you should not add SYSID= to an existing APPLCTN macro for a MODBLKS generation. Doing so causes the local application to be deleted after an online change sequence of commands.

A PSB cannot be initially defined or redefined from remote to local by using the SYSID= parameter on the APPLCTN macro during an online change.

The SYSID keyword parameter is invalid if FPATH=YES is specified.

SCHDTYP=

Specifies whether (PARALLEL) or not (SERIAL) this application program can be scheduled into more than one message region or batch message region simultaneously. The default value is SERIAL.

When the SCHDTYP= parameter is changed by a MODBLKS system definition from serial to parallel or vice versa, and the PSB, defined as resident, is not changed, the PSB is considered non-resident until the next IMS restart.

SCHDTYP= and the positional parameter DOPT are mutually exclusive.

BUFPOOLS Macro

The BUFPOOLS macro statement is used to specify default storage buffer pool sizes for the DB/DC and DBCTL environments. The sizes specified are used unless otherwise expressly stated for that buffer or pool at control program execution time for an online system.

The following keywords are no longer supported:

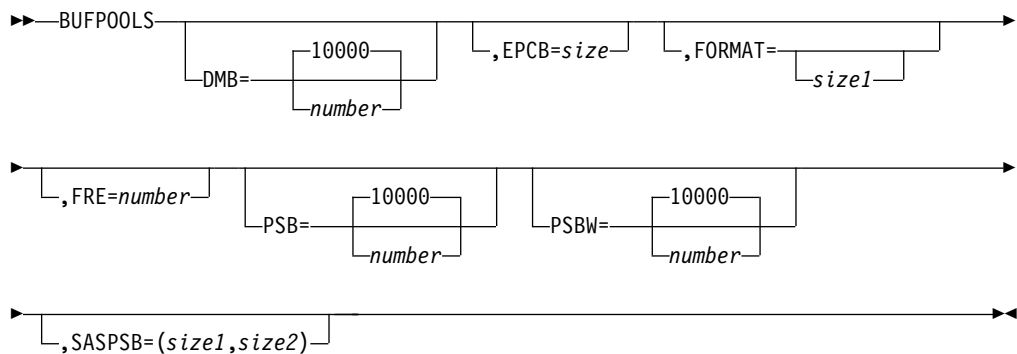
- The DBASE keyword.
You now specify the common database buffer pool size using DFSVSAMP data set parameters.
- The COMM keyword.
- The GENERAL keyword.

BUFPOOLS Macro

Values entered for these keywords are ignored. The DBASE keyword, which specified the size of the common database buffer pool, is no longer supported. The common database buffer pool size is now specified using DFSVSAMP data set parameters. See “Specifying IMS Buffer Pools” on page 378 for more information on common database buffer pool definition.

The COMM keyword, which specified additional space to be allocated to the CIOP pool, is no longer supported. The GENERAL keyword, which specified the size of the SPAP pool, is also no longer supported. Values entered for either of these keywords are ignored.

The Work Area Pool (WKAP) cannot be specified at system definition time. System definition sets a value of 5000. You can override this at execution time by using the WKAP= operand in the IMS, DBC, or DCC procedures, or in the DFSPBIMS, DFSPBDBC, or DFSPBDCC members.



Keyword Parameters

To find which parameters apply to your IMS configuration, refer to Table 12 on page 32.

DMB=

Specifies the size of the DMB control block pool. The default is 10000 bytes. The maximum allowable specification is 9999000 bytes. The minimum allowable specification is 8 bytes.

EPCB=

Specifies the size of the EPCB pool. If Fast Path is generated, the default size is 8 KB. Otherwise the default EPCB pool size is 0. The maximum allowable specification is 9999000 bytes. The storage for this pool resides in ECSA.

The following should be calculated for MPP, IFP, and BMP region types that use Fast Path resources (DEDBs, MSDBs, or EMH).

Table 15. Calculating EPCB Storage for PSB

EPCB Type	Size (in bytes)	Total VS Requirements for This EPCB Type	Total
IOPCB	28 b	28	_____K
Alt-Resp	28 b	28 x (# Alt-Resp PCBs)	_____K
MSDB	76 b	76 x (# of MSDB PCBs)	_____K
DEDB	* b	Sum of all DEDB PCBs	_____K
Total EPCB storage for this PSB			_____K

Formula for Calculating Value of EPCB Pool:

132 + (124 x # of SENSEG statements
in the DEDB PCB).

After working this calculation for each PSB, determine the largest PSB that might execute in a region, and add up all the results for all regions. This is the value for your EPCB pool. This pool specification can be overridden via the IMS procedure using the EPCB parm.

FORMAT=

Specifies sizes of the message format block pool.

size1 specifies the size of the message format block pool. The minimum specification is 2000 bytes; the maximum specification is 99999000 bytes. This parameter is ignored when no unit types requiring MFS are included in the defined system. If unit types that require MFS are included in the defined system and this parameter is not specified, the default value is calculated. *size2* is no longer supported.

The default value is calculated as follows:

$$1000 + X + 2124 + FRE \times 44$$

where:

X Is the greater of: $((3270B/2) + (3270V/8)) \times 2200$ or 14336

3270B Is the number of 3270 BTAM lines

3270V Is the number of 3270 VTAM terminals

FRE is the number of specified or defaulted FREs

FRE=

Specifies the number of fetch request elements (FRE) for loading MFS control blocks into the message format block pool. The number specified can range from 10 to 99999. If the size of the buffer pool is increased, the number of FREs should also be increased. Otherwise, the additional space cannot be used.

The default value is calculated as follows:

$$(10 + X) \text{ or } 30, \text{ whichever is greater}$$

where:

X $((3270B/2) + (3270V/8)) \times 4$

3270B Is the number of 3270 BTAM lines

3270V Is the number of 3270 VTAM terminals

PSB=

Specifies the size of the PSB control block pool if the DL/I address space option is not used. The SASPSB parameter specifies the size of the PSB control block pool when the DL/I address space option is used. The default is 10000 bytes, with a maximum of 9999000 bytes. The minimum allowable specification is 8 bytes.

PSBW=

Specifies the size of the PSB work area pool. The default is 10000 bytes. The maximum allowable specification is 9999000 bytes. The minimum allowable specification is 8 bytes.

BUFPOOLS Macro

SASPSB=

Is used only if the DL/I subordinate address space option is selected. If you are not using this option, the size of the single PSB control block pool is specified with the PSB parameter.

With the DL/I address space option, two PSB control block pools exist. *Size1* is the size of the pool in the MVS CSA. *Size2* is the size of the pool in DL/I local storage. The maximum allowable for either is 9999000 bytes.

Both *size1* and *size2* must be specified. Normally, the value of *size2* should be larger than that of *size1*, and neither value can be 0.

The ACBGEN utility output provides information on the relative PSB pool sizes. You should examine the output of ACBGEN before coding SASPSB.

The defaults are:

size1 20% of the PSB specification on BUFPOOLS

size2 80% of the PSB specification on BUFPOOLS

For example, SASPSB=(20000) is invalid, because *size2* is missing. The defaults apply only if the SASPSB parameter is not specified.

Example: The following example shows how to code BUFPOOLS. If the DL/I address space option is not used, the single PSB pool size is 80000 bytes. If the option is selected, the PSB pool size in the CSA is 16000 bytes, and the DL/I local storage pool size is 64000 bytes. This example also specifies a 40000-byte DMB pool.

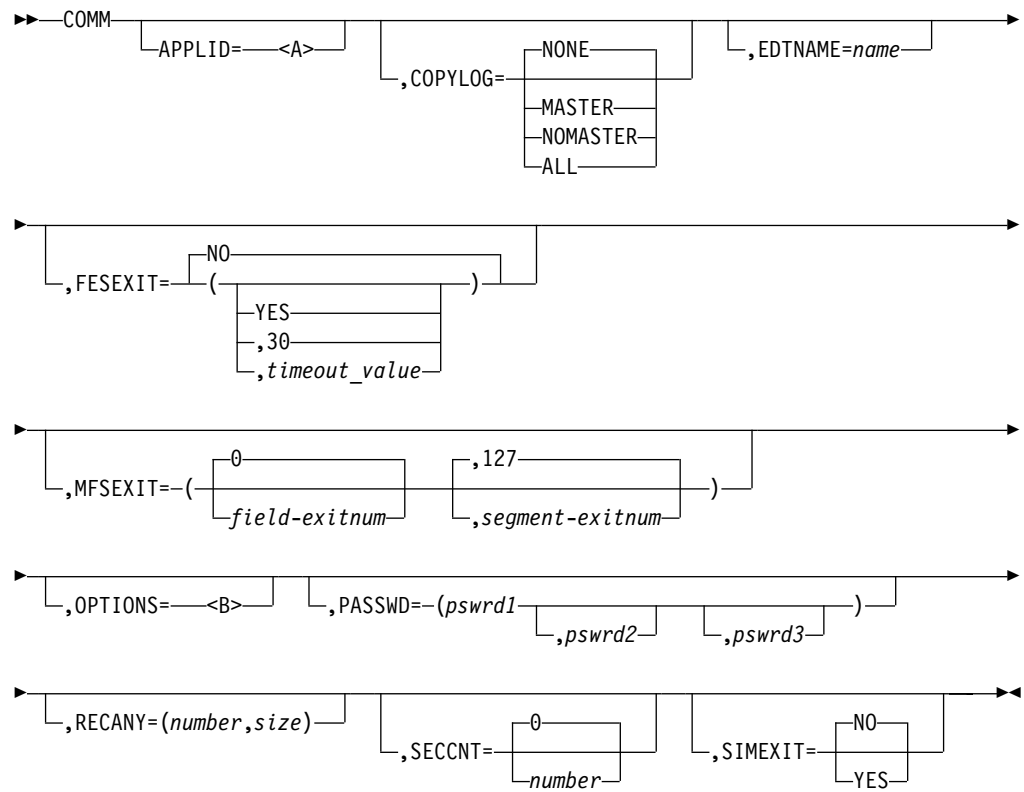
```
BUFPOOLS PSB=80000,SASPSB=(16000,64000),  
DMB=40000
```

COMM Macro

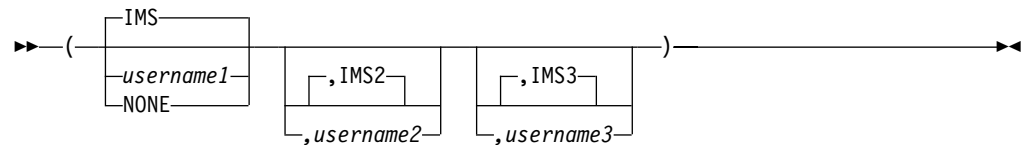
The COMM macro is used to specify general communication requirements that are not associated with any particular terminal type. COMM is always required for terminal types supported by VTAM. It is optional for BTAM, BSAM, GAM, and ARAM terminal types. It can also be required to specify additional system options, such as support for MFS on the master terminal.

The COMM macro should be placed prior to the data communication specifications in the stage 1 input sequence. Refer to "Appendix B. MFS System Definition and Programming Considerations" on page 461 for specific examples.

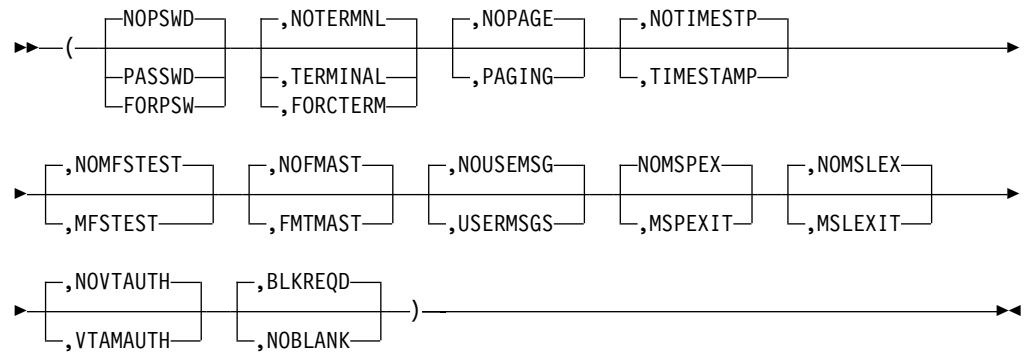
COMM Macro



<A> is



 is



Keyword Parameters

To find which parameters apply to your IMS configuration, refer to Table 12 on page 32.

COMM Macro

AOEXIT=

No longer supported.

APPLID=

Specifies the application identification in the VTAM ACBs if VTAM is used. The default value is IMS. When XRF is not used, you should specify only one VTAM APPLID.

For XRF, you should specify two VTAM APPLIDs *username1* and *username2*. These two parameters correspond to the two XRF systems. The defaults are IMS and IMS2, respectively.

For RSR without XRF, you should specify at least two VTAM APPLIDs: *username1* (or *username2*) and *username3*, where *username3* is the application identification for the RSR tracker. In all cases, *username3* can only be used by the RSR tracker.

For RSR with XRF, you use all three VTAM APPLIDs: *username1* is for your active IMS (for both XRF and RSR), *username2* is your XRF alternate (at the RSR active site), and *username3* is your RSR tracker.

All three username values can be overridden by specifying APPLID1=, and either APPLID2= or APPLID3=, or both, in the IMS or DCC procedures. These keywords can also be specified in the DFSPBIMS or DFSPBDCC members. *Username3* can match either *username1* or *username2*, but no two APPLIDs can be active in the network at the same time.

The defaults are IMS, IMS2, and IMS3, respectively.

If NONE is specified:

- The MSVERIFY program (DFSUMSV0) cannot validate that the APPLID names of the IMS system and its partner match if MSC/VTAM links are defined.
- Either the job step name of the job to bring up the IMS control region is used, or the name of the started task (if a started task is used for the IMS control region) is used.

If you specify APPLID=NONE and you issue a /DIS ACTIVE DC command, the output of the command is APPLID=NONE.

APPLID=NONE cannot be specified for XRF or for RSR.

COPYLOG=

Specifies the different categories of command and command response messages for which a copy is to be printed on the secondary master terminal.

If you want hardcopy logging, you must define a secondary master terminal.

The default is NONE.

NONE Specifies that the commands listed in Table 16 on page 67 and command responses from terminals are not to be printed on the secondary master terminal. The queuing for the secondary master terminal of messages not in the categories described in Table 16 on page 67 below is not affected by the COPYLOG statement.

MASTER

Specifies that copies of commands listed in Table 16 and command responses from the master terminal are to be printed on the secondary master terminal.

NOMASTER

Specifies that copies of the commands listed in Table 16 below and command responses from terminals other than the master terminal are to be printed on the secondary master terminal.

ALL

Specifies that the commands listed in Table 16 and command responses from all terminals are to be printed on the secondary master terminal.

COPYLOG has no effect on system messages, which continue to be logged to the secondary master terminal.

The commands that can be copied to the secondary master terminal are:

Table 16. Commands That Can Be Copied to the Secondary Master Terminal

/ACTIVATE	/MSASSIGN
/ASSIGN	/OPNDST
/CHECKPOINT	/PSTOP
/CLSDST	/PURGE
/COMPT	/QUIESCE
/DBDUMP	/RCLSDST
/DBRECOVERY	/RCOMPT
/DELETE	/RSTART
/DEQUEUE	/START
/DISPLAY	/STOP
/IDLE	/TRACE
/MONITOR	/UNLOCK SYSTEM

Other commands can be logged but are not under the control of COPYLOG as /MSVERIFY is, for example.

The COPYLOG specification can be dynamically changed at execution time by the /SMCOPY operator command.

EDTNAME=

Names the ISC edit process for LU 6.1 nodes. The name specified becomes a synonym for ISCEDT, the default name that can be input and is output on the DPN parameter of the SCHEDULER function management header. When this name is input to IMS on the DPN parameter of the ATTACH or SCHEDULER function management header, it causes IMS to use the ISC edit process. The name used on the EDTNAME parameter cannot be the same as any MFS message input descriptor (MID) name used as an input ATTACH DPN parameter to invoke an MFS process.

FESEXIT=

Specifies whether (YES) or not (NO) the Front End Switch exit routine is to be included in the IMS nucleus.

The default is NO.

The *timeout_value* specifies a timeout value in seconds. A *timeout_value* specifies the time interval in which a reply for an FES message must be

COMM Macro

received. If no reply is received by the front-end system in time, timeout processing begins. The minimum value is 1, and the maximum value is 300.

If FESEXIT=YES is specified without a timeout value, the default for the *timeout_value* parameter is 30.

The FES exit routine must be placed in the data set defined by the USERLIB= keyword (of the system definition IMSGEN macro) prior to the execution of stage 2. The name of the exit routine must be DFSFEBJ0.

FESEXIT=(YES) and OPTIONS=NOBLANK are mutually exclusive. If you specify both, stage one assembly will not complete successfully.

Related Reading: More information on the Front End Switch exit routine can be found in *IMS/ESA Customization Guide*.

MFSEXIT=

The variable *fieldexitnum* specifies the highest-numbered MFS Field Edit exit routine to be included in the generated system. *fieldexitnum* must be a decimal number greater than or equal to 0 and less than the default or specified value for the *segmentexitnum* parameter. The default is 0.

The *segmentexitnum* value specifies the lowest-numbered MFS Segment Edit exit routine to be included in the generated system. The *segmentexitnum* must be a decimal number greater than the default or specified value for the *fieldexitnum* parameter and less than or equal to 127. The default is 127.

Recommendation: Make lower-numbered exit routines be field exit routines, even though the lower-numbered exit routines can be segment exit routines, or can be a mixture of field and segment exit routines. An additional recommendation is that the higher-numbered exit routines be segment exit routines, although the higher-numbered exit routines can be field exit routines, or can be a mixture of field and segment exit routines.

An MFSEXIT=(14,120) specification indicates that user-supplied MFS Field Edit exit routines DFSME001 through DFSME014 and user-supplied MFS Segment Edit exit routines are IMS-supplied routines and are always included in the generated system when MFS requirements exist.

User-supplied MFS exit routines are included in the system definition-generated online control blocks module DFSIBLKx. Therefore, MFS exit routine specifications can be changed via an IMS CTLBLKS, NUCLEUS, ON-LINE, or ALL generation. The MFSEXIT keyword specification is ignored when MFS exit requirements do not exist in the generated system.

OPTIONS=

The parameters described below are not position dependent within the operand sublist. If any terminal or password security options are specified here and also in the IMSGEN or SECURITY macro statement, a warning message will be issued.

NOPSWD | PASSWD | FORPSW

Specifies whether (PASSWD or FORPSW) or not (NOPSWD) password security is to be generated as the default for the /NRESTART command when issued to start IMS. This parameter value is ignored if the SECURITY macro is specified, and it is included for compatibility purposes only. Default

value is NOPSWD. See “Relationships between the SECURITY, COMM, and IMSGEN Macros” on page 137 for more information.

FORPSW specifies that the IMS master terminal operator cannot negate password security when issuing the /NRESTART command. If PASSWD is specified, the MTO can override password security inclusion by specifically requesting no security on the /NRESTART command.

NOTERMNL | TERMINAL | FORCTERM

Specifies whether (TERMINAL or FORCTERM) or not (NOTERMNL) terminal security is to be generated as the default for the /NRESTART command when issued to start IMS. This parameter will be ignored if the SECURITY macro is specified, and is included for compatibility purposes only. The default is NOTERMNL. See “Relationships between the SECURITY, COMM, and IMSGEN Macros” on page 137 for more information.

FORCTERM specifies that the IMS MTO cannot negate terminal security when issuing the /NRESTART command. If TERMINAL is specified, the master terminal operator can override terminal security by specifically requesting no security on the /NRESTART command.

NOPAGE | PAGING

Defines whether (PAGING) or not (NOPAGE) the terminal paging feature is included in the defined system. The default is NOPAGE. If 3270 or SLU 2 devices are included, or if SLU P, LU 6.1, or 3600 devices defined with MFS are included, the paging capability is automatically included.

NOTIMESTP | TIMESTAMP

Specifies whether (TIMESTAMP) or not (NOTIMESTP) the system message time stamp feature is desired in the defined system. The default is NOTIMESTP.

When TIMESTAMP is specified, the time at which a message was generated is inserted between the message number and the message text for each message in the table.

Related Reading: Refer to the DFS messages introduction in *IMS/ESA Messages and Codes* for more information.

NOMFSTEST | MFSTEST

Specifies whether (MFSTEST) or not (NOMFSTEST) the Message Format Service test facility, MFSTEST, is to be included in the generated system. Specifying MFSTEST is invalid for systems not containing MFS terminals. The default is NOMFSTEST.

Specifying a numeric value on the COMM macro statement is no longer recommended; for compatibility reasons it is still allowed, but any numeric value is ignored. When a numeric value is specified the NOMFSTEST default is overridden and MFSTEST is set; if MFSTEST=NO is specified on the IMSGEN macro, it is also overridden.

Using MFSTEST can degrade IMS performance due to MFSTEST use of the CIOP pool.

NOFMAST | FMTMAST

Specifies whether (FMTMAST) or not (NOFMAST) the IMS-provided

COMM Macro

support for MFS on the master terminal is to be used. This support is available for the following devices used as the master terminal:

- 3277 model 2
- 3270 displays that have a 24 x 80 screen and use symbolic names
- 3277 model 2 when defined as SLUTYPE2

NOUSEMSG | USERMSG

Specifies whether (USERMSG) or not (NOUSEMSG) the user-supplied user message tables module DFSCMTU0 is to be included within the generated system.

NOMSPEX | MSPEXIT

Specifies whether (MSPEXIT) or not (NOMSPEX) the user-written MSC Program Routing exit routine (DFSCMPRO) is to be included in the generated system.

NOMSLEX | MSLEXIT

Specifies whether (MSLEXIT) or not (NOMSLEX) the user-written MSC Link Receive Routing exit routine (DFSCMLR0) is to be included in the generated system.

NOVTAUTH | VTAMAUTH

Specifies whether (VTAMAUTH) or not (NOVTAUTH) IMS is to use the VTAM authorized path for communicating.

Related Reading: Refer to *ACF/VTAM Macro Language Guide* for information about the authorized path facility. These parameters can be overridden by the JCL prepared for system execution.

BLKREQD | NOBLANK

Specifies whether the one- to eight-byte transaction code requires a trailing blank. NOBLANK lets you enter a one- to eight-byte transaction code without having to specify a trailing blank.

OPTIONS=NOBLANK and FESEXIT=(YES) are mutually exclusive. If you specify both, stage one assembly does not complete successfully.

PASSWD=

Specifies, when neither XRF nor RSR is used, that one password is to be specified in the VTAM ACB. This password is checked by VTAM. If no password is specified, and VTAM requires one (during VTAM system generation), the IMS VTAM ACB is not initialized.

For XRF, you should specify two VTAM passwords that correspond to the two XRF systems. If only one password is specified, it is used as the password for both XRF systems.

For RSR without XRF, you should specify at least two VTAM passwords:

- *password1* (or *password2*)
- *password3*

In all cases, *password3* can only be used by the RSR tracker.

For RSR with XRF, you will use all three VTAM passwords:

- *password1* for your active IMS (for both XRF and RSR)
- *password2* for your XRF alternate (at the RSR active site)
- *password3* for your RSR tracker

If you specify only one password, it is used for all systems. If you specify only two passwords, *password1* is used for the RSR tracker.

RECANY=

This parameter is required only when defining VTAM terminals. It defines the number and size of the RECEIVE ANY buffers.

This keyword and its parameters are optional and can be either overridden with execution-time parameters or defaults can be used if not specified by either of these methods.

number

Specifies the number of VTAM RECEIVE ANY buffers to be present in the IMS system. Valid values are from 1 to 500. The default is 16 if not specified during system definition or with execution-time parameters. You can override this specification with the EXEC parameter RECA= in the IMS procedure.

Specifying too many buffers does not improve performance and can result in excessive resource consumption, especially high CPU utilization. You can check current and maximum use by looking in the X'450D' log record.

size

Specifies the size of the largest RECEIVE ANY buffer. This size must be large enough to handle maximum input from any VTAM-attached terminal. The minimum acceptable size is 50; the maximum is 30720. The default is 1920 if not specified during system definition or with execution-time parameters. The resulting usable input buffer size for any terminal can be calculated as follows for the devices shown:

3601: Size is record size plus header. The header size is 2 bytes if MFS is not used and 11 bytes if MFS is used.

3270: Size is the input data stream length. The minimum size is 300 bytes and a reasonable maximum size is 3842 bytes.

SLU 1, SLU 2: Record size should be equal to or greater than the largest input record to be received.

SLU P: Size is record size plus header. The header size is 5 bytes if MFS is not used and variable from 7 to 40 bytes if MFS is used.

LU 6.1: Size is record size plus header. The header size without MFS is variable from 0 to 45 bytes. The header size with MFS is variable from 17 to 52 bytes.

NTO: Record size should be equal to or greater than the largest input record to be received.

For all VTAM terminals, the RECANY buffer size must be a decimal value that can be expressed by the algorithm $X \text{ times } 2 \text{ to the power of } Y$. X must be a value from 8 through 15, and Y must be a value from 3 through 11. This is a VTAM restriction. Thus, for example, the value 144 (representing 9×2^4) or the value 28672 (representing 14×2^{11}) are acceptable values.

Related Reading: Details of this VTAM restriction are in *ACF/VTAM Macro Language Reference*.

COMM Macro

This restricts the valid VTAM buffer sizes to one of the following decimal values:

112	480	2048	9216
120	512	2304	10240
128	576	2560	11264
144	640	2816	12288
160	704	3072	13312
176	768	3328	14336
192	832	3584	15360
208	896	3840	16384
224	960	4096	18432
240	1024	4608	20480
256	1152	5120	22528
288	1280	5632	24576
320	1408	6144	26624
352	1536	6656	28672
384	1664	7168	30720
416	1792	7680	
448	1920	8192	

IMS converts the specified decimal value to the format required for the bind parameter fields and places the value into the appropriate field in the bind parameter list. If the maximum request unit (RU) value specified cannot be converted exactly into the bind format, the value is rounded down to the next-lower value bind format for inbound RUs and rounded up to the next-higher value bind format for outbound RUs.

SECCNT=

Specifies the maximum number of terminal and password security violations to be accepted per physical terminal prior to master terminal notification of such violation. The default is 0, which nullifies notification of the master terminal. The number specified can be 0, 1, 2, or 3. This value is reset to 0 upon successful signon and does not continue across transactions.

If SECCNT is not 0, the master terminal is notified for every violation.

SIMEXIT=

Specifies whether (YES) or not (NO) the Shared Printer Message Router exit routine is to be included in the IMS nucleus.

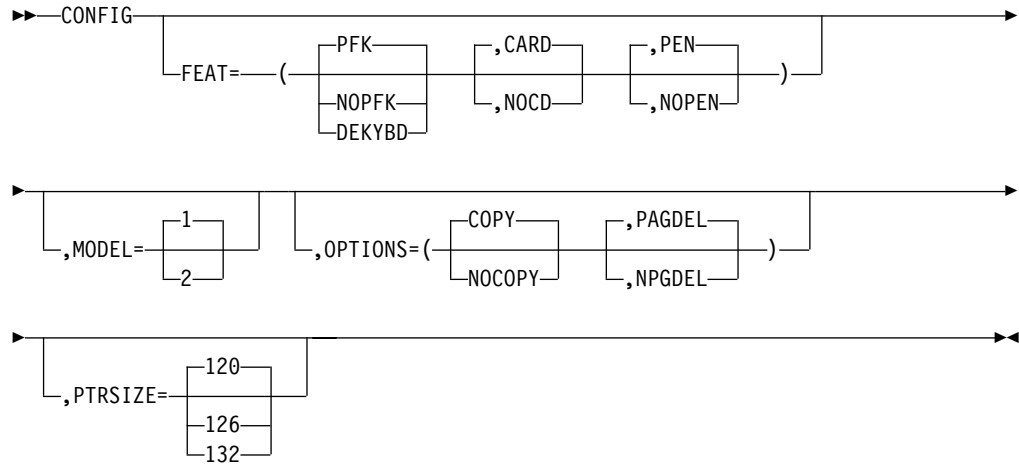
This exit routine must be placed in the data set defined by the USERLIB= keyword of the system definition MSGEN macro statement prior to the execution of stage 2. The name of the exit routine must be DFSSIML0. A default exit routine is not provided.

The exit routine is bypassed unless SIMEXIT=YES is specified. If SIMEXIT=YES is specified, the exit routine DFSSIML0 is called when a message is queued for terminals defined with OPTIONS=SHARE on the TERMINAL macro statement.

When SIMEXIT=NO, the default, the router module assumes that /OPN is always simulated when output is enqueued for a terminal defined as OPTIONS=SHARE on the TERMINAL macro statement. Even if SIMEXIT=NO is specified, the module DFSSIML0 can be manually link-edited into the nucleus.

CONFIG Macro

The CONFIG macro statement provides the configuration for a switched 3275 terminal. Because the configuration provided by CONFIG is referenced when the named 3275 dials into IMS, differently configured 3275s can use the same communication line. All CONFIG macro statements must be between the LINEGRP macro and the LINE macros. LINE macros can refer to named CONFIG macros defined previously in this line group or in previously defined line groups.



Label Field

Specifies a one- to eight-character alphanumeric name. This CONFIG name can be referenced in subsequent LINE and IDLIST macro instructions.

Keyword Parameters

To find which parameters apply to your IMS configuration, refer to Table 12 on page 32.

FEAT=

Describes this 3275:

- The first parameter specifies whether program function keys (PFK), no program function keys (NOPFK), or a data-entry keyboard (DEKYBD) is available. The default is PFK.
- The second parameter specifies whether (CARD) or not (NOCD) the operator identification card reader is available. The default is CARD.
- The third parameter specifies whether (PEN) or not (NOPEN) the selector light pen is available. The default is PEN.

MODEL=

Specifies the terminal model number. The default is 1.

OPTIONS=

Specifies operational options for this 3275:

- The first parameter specifies whether (COPY) or not (NOCOPY) the copy function is requested. The default is COPY.
- The second parameter specifies whether (PAGDEL) or not (NPGDEL) the automatic page deletion function is requested. The default is PAGDEL.

PTRSIZE=

Specifies the platen size of the attached 3284 Model 3. The default is 120.

CONFIG Macro

Example: The following example provides the configuration for a switched 3275 Model 2 with a printer platen size equal to 120. The features available are data entry keyboard and selector light pen. The copy and automatic page deletion functions are requested.

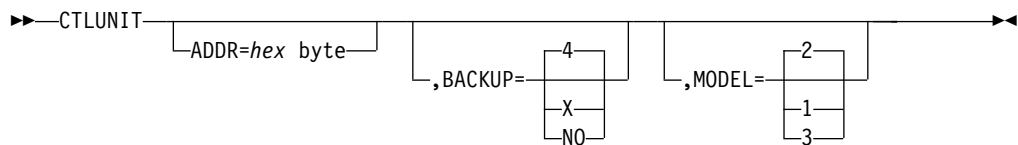
```
C1 CONFIG MODEL=2,FEAT=(DEKYBD,NOCD)
```

CTLUNIT Macro

The CTLUNIT macro statement specifies 2848, 2972, and 3271 control unit characteristics.

- For 2848 control units, CTLUNIT specifies the control unit address, write line addressing (WLA) feature, and 2848 model number.
- For 2972 control units, CTLUNIT specifies control unit address and buffer size.
- For 3271 control units, CTLUNIT specifies control unit address and model number.

CTLUNIT is valid only for 3270 remote line groups.



Keyword Parameters

To find which parameters apply to your IMS configuration, refer to Table 12 on page 32.

ADDR=

Specifies the two-digit hexadecimal address of the 3271 associated with the remote 3277, or the two-digit hexadecimal address of the 2972 control unit. The address was assigned by the IBM customer engineer upon installation of the control unit. For the 3271, the IBM customer engineer assigns the selection address, which must be converted to the polling address for specification here. To be acceptable as a valid 2972 control unit address, this operand must be the hexadecimal representation of a character from A through Z.

BACKUP=

Specifies (only for XRF) the control of the automatic restart after takeover; it is used only when HSB=YES is specified in the IMSCTRL macro.

X is a numeric integer from 1 (lowest) to 7 (highest), inclusive, that specifies the priority assigned to reestablishing the session.

When either the keyword or the parameter is omitted, the default is 4. If BACKUP=NO is specified, the automatic restart of the devices is suppressed.

MODEL=

Specifies the 2848 or 3271 control unit model number.

For 3271 control units only, the value 1 or 2 can be specified. The default is 2.

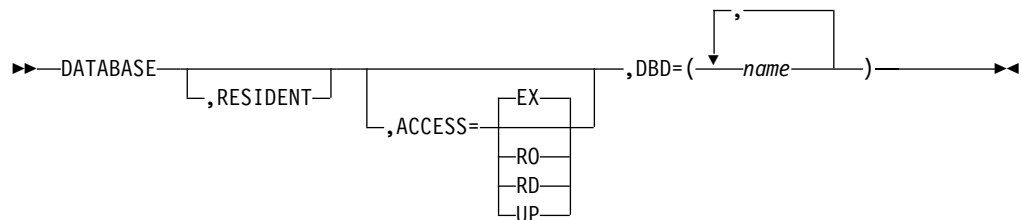
For 3274-1C control units, only the value 2 can be specified.

For 2848 control units, the entered value determines the input buffer size for terminals attached to this control unit. The default value is 3.

DATABASE Macro

The DATABASE macro statement is used to define the set of physical databases that IMS is to manage. One DATABASE macro instruction must be specified for each HSAM, HISAM, and HDAM database. Two DATABASE macro instructions are required for a HIDAM database: one for the INDEX DBD and one for the HIDAM DBD. One DATABASE macro instruction must be included for each secondary index database that refers to any database defined to the online system.

For Fast Path, a DATABASE macro statement must be included for each Main Storage Database (MSDB) and Data Entry Database (DEDB) to be processed.



Positional Parameter

RESIDENT=

Indicates that the DMB directory control block created for this DATABASE statement should reside in storage during system and DBCTL initialization.

Fast Path makes the database control blocks resident in storage during system initialization regardless of whether the RESIDENT option is specified.

Online changes to the RESIDENT option are permitted. However, DMBs associated with new databases defined as RESIDENT are not made resident until the next IMS restart. Until that time, they are treated as nonresident. Changes to a RESIDENT DMB (from an ACBGEN) cause that DMB to be treated as nonresident until the next restart of IMS.

Keyword Parameters

ACCESS=

When used in conjunction with the database-sharing level declared to database recovery control (DBRC), this parameter specifies the access for the defined database, that is, how the subsystem requesting access plans to use the database.

The ACCESS parameter can be modified by a /START command during IMS execution. Because this parameter can be changed using a /START DATABASE command, online changes to the ACCESS keyword have no effect until the next cold start.

You cannot specify an ACCESS value for MSDBs and GSAM. The possible ACCESS values are:

EX (exclusive)

Indicates that the named database can be accessed exclusively by this IMS subsystem. This is the default for ACCESS. Concurrent access to this

DATABASE Macro

database by other subsystems is prohibited by DBRC. Unless this is a DEDB, PCBs that refer to this database are scheduled, regardless of the PROCOPT values specified in PSBGN. If this is a DEDB, PROCOPT=EX is not permitted.

Concurrent image copy cannot be run against a database with ACCESS=EX. If you are running with high-speed sequential processing (HSSP), you must specify ACCESS=EX or ACCESS=UP.

RD (read)

Indicates that the named database can only be read by this subsystem. The database is opened for input only. No application programs are allowed to physically update this database.

RO (read only)

Indicates that the named database is read-only in this IMS subsystem. This database is opened for input only. PCBs that refer to this database are allowed only if PROCOPT=GO is specified in PSBGEN.

UP (update)

Indicates that the named database can be updated and read. PCBs that refer to this database can be scheduled regardless of their PROCOPT values.

If you are running with high-speed sequential processing (HSSP), you must specify ACCESS=UP or ACCESS=EX.

Related Reading: To understand the interaction between the ACCESS parameter on the DATABASE macro and the DBRC and IRLMNM parameters on the IMSCTRL macro statement, see *IMS/ESA Administration Guide: System*.

DBD=

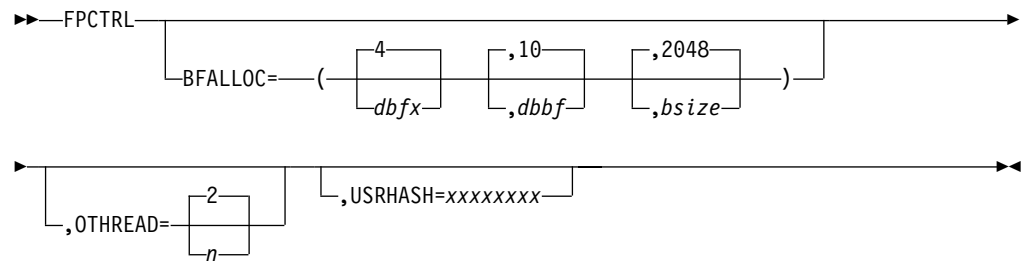
Specifies the name of one or more database descriptions (DBDs). If more than one DBD name is specified, each is assumed to have the same characteristics as any positional parameter specified or used by default. The first character of the name must be a valid alphabetic character (A through Z, #, \$, or @). At execution time, the DBD must have been processed by the block builder utility program. It must exist as a member in the partitioned data set named in the IMSACB DD statement (IMS.ACBLIB). If the DBD has not been processed by the block builder utility program, and is, therefore, not present in the ACBLIB data set, the database is locked at execution time. This operand is required.

Example:

```
DATABASE DBD=(N1,N2,N3)
```

FPCTRL Macro

The FPCTRL macro statement defines the IMS Fast Path options of the IMS control program, and the DBCTL environment. This statement must appear somewhere between the IMSCTRL and IMSGEN macro statements if Fast Path is used. It is ignored when the IMSCTRL statement specifies that only a BATCH or MSVERIFY system definition is to be performed. If Fast Path is to be included in a DCCTL environment, specify the FPCTRL macro without any parameters.



Keyword Parameters

To find which parameters apply to your IMS configuration, refer to Table 12 on page 32.

Related Reading: If Fast Path is implemented in your IMS configuration, see “Member DFSFIXnn” on page 309 for Fast Path page fix recommendations.

BFALLOC=

Specifies the values used to allocate a storage pool from which storage for control blocks and buffers associated with the Data Entry Database (DEDB) and Main Storage Database (MSDB) is obtained. The default values are used unless these are overridden by EXEC statement parameters when an IMS online control region is executed. See “Specifying IMS Execution Parameters” on page 365.

For an RSR tracking subsystem, these database buffers are used as I/O staging buffers. Also for RSR, be sure to convert your MSDBs to DEDB areas or DEDB VSO areas.

Related Reading: See *IMS/ESA Utilities Reference: Database Manager*.

dbfx

Specifies additional buffers (not separate pools) to be page-fixed at the start of the first Fast Path region. This specification allows for asynchronous processing where the DEDB updates are held until the associated log buffer is written. The default is 4.

This parameter is not used by an RSR tracking subsystem.

dbbf

Specifies the total number of database buffers. The specification should satisfy the following formula:

dbbf must be equal to or greater than the sum of open DEDB areas that have sequential dependent segments defined, plus:

- The sum of all Fast Path buffers used by CCTLs (CNBA)
- The sum of all database buffers (NBAs) for any executing Fast Path-dependent region
- The *dbfx* specification
- The maximum overflow buffer allowance (OBA) for any executing Fast Path region

FPCTRL Macro

The valid range is 1 to 65535. If you intend to use a DEDB or MSDB, 0 is not reasonable. Insufficient allocation of buffers can result in an open error for an area or in a start error for a region. The default is 10.

bsize

Specifies the size of the database buffers. It must be equal to or greater than the largest control interval size used by the DEDB. The only valid values are 512, 1K, 2K, 4K bytes, and multiples of 4K bytes up to 28K bytes. The default is 2048 bytes (2K bytes).

INREXIT=

Is no longer used. An exit routine DBFHAGU0 is automatically included for Fast Path. If you were using the DBFSUIX0 sample exit routine, the new sample exit routine DBFHAGU0 should be compatible.

Related Reading: See the *IMS/ESA Customization Guide* for more information.

OTHREAD=

Specifies the number of concurrent output threads that IMS Fast Path is to permanently allocate. For an RSR tracking subsystem, the OTHREAD= parameter is used to determine the maximum number of concurrent I/O operations for Fast Path database tracking.

The valid range is from 1 to 255, with a default of 2. Each output thread is dispatched in the control region and performs the output functions for the Data Entry Database (DEDB).

USRHASH=

If this parameter is specified in the FPCTRL macro, it will be ignored.

Example: The following is an example of the FPCTRL macro statement.

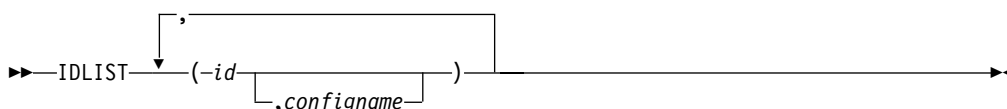
```
FPCTRL OTHREAD=3,BFALLOC=(30,100,1024)
```

IDLIST Macro

The IDLIST macro statement is used to create a terminal security list for switched 3275s. Each entry consists of a 3-byte hexadecimal value (derived from the last 3 bytes of the terminal's 4-byte hardware ID) and the name of a previously defined configuration. The maximum number of ID entries—*(id,configname)* is an entry—that can be specified in an IMS system definition is 5000. If the IDLIST macro is used, it must follow the LINE macro.

If the IDLIST macro is not used, the list specified by IDS= in the preceding LINE macro is used to determine the 3275 configuration. If IDS= is not specified, the terminal configuration used is the same for all terminals that dial in on this line (either the CONFIG= specified in the LINE macro or the IMS default).

At least one IDLIST macro is required if VERIFY=YES is specified on any LINE macro. A LINE macro that specifies VERIFY=YES must either also specify IDS= or be followed by an IDLIST macro.



Label

A one- to eight-byte alphanumeric name, which the IDS= operand of a subsequent LINE macro can reference.

Positional Parameters

To find which parameters apply to your IMS configuration, refer to Table 12 on page 32.

id Specifies in hexadecimal the last three bytes of the terminal's 4-byte hardware ID.

configname

Specifies the name of the configuration to be used for this terminal. If *configname* is specified, the name provided must reference a previously defined CONFIG macro. If *configname* is not specified, the configuration named in the CONFIG= operand of the associated LINE macro is used. If no configuration is specified in the LINE macro, the IMS default is used.

Example: The following is an example of the IDLIST for four switched 3275s:

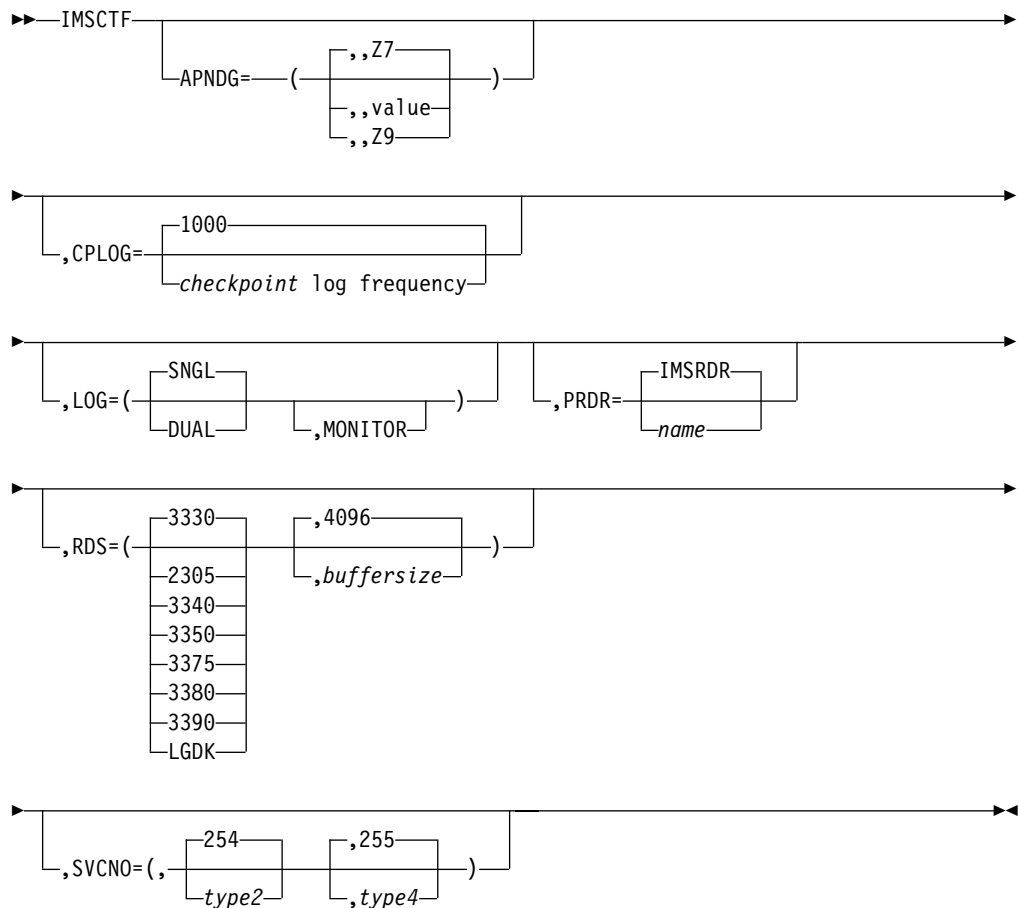
```
IDLIST (E54C5A,A),(E54C5B),(E54C5C,C2),(E54C5D)
```

The terminal IDs are E54C5A, E54C5B, E54C5C, and E54C5D. Terminal E54C5A uses the configuration described by the CONFIG macro named A. Terminal E54C5C uses the configuration described by the CONFIG macro named C2. Terminals E54C5B and E54C5D use either the configuration named in the LINE macro or the IMS default configuration.

IMSCTF Macro

The IMSCTF macro statement defines parameters to IMS, and to the DBCTL environment. The DYLOG and DISKLOG keywords and the third parameter (2|number) on the RDS keyword are no longer used and are not included in the macro format below. If either of these are specified, they are ignored. They can remain in your system definition to provide compatibility with previous IMS releases.

IMSCTF Macro



Keyword Parameters

To find which parameters apply to your IMS configuration, refer to Table 12 on page 32.

APNDG=

The first operand and second operands are no longer used. The comma is retained to maintain compatibility with previous releases.

The third operand specifies the Multiple Systems Coupling channel-to-channel appendage suffix for MVS/ESA systems. The value specified can range from WA to Z9 (the default is Z7).

User appendages IGG019Z7 or IGG019Z8 can already exist. If any of these appendages do exist, you should do one of the following:

- Rename the existing appendages
- Do not permit the APNDG parameter to default to Z7, but instead choose other suffixes

CORE=

No longer used. CORE= has been replaced by the PIINCR and PIMAX execution parameters for DB/DC and DBCTL IMS online systems, respectively.

CPLOG=

Specifies the number of system log records between system-generated checkpoints. The permitted value ranges from 500 to 16777215. The default is 1000.

LOG=

The first operand, SNGL|DUAL, is no longer used for an online execution. If specified in an online configuration, it is ignored.

When specified for a batch execution, this parameter determines whether 1 (SNGL) or 2 (DUAL) DD statements are to be generated in the IMS batch procedures for the system log.

The second parameter specifies that the IMS Monitor logging is also to be included.

PRDR=

Specifies an optional IMS reader proc name. The name must be one- to eight-alphanumeric or national characters, with the first being alphabetic or national. The default is IMSRDR. You can also specify the reader proc name in the control region JCL and in the default parameter module, DFSPBIMS. DBCTL users use DFSPBDBC.

RDS=

The first parameter specifies the device type on which the restart data set (IMS.RDS) is to reside.

You can specify one of the following device types: 3330 (the default), 2305, 3340, 3350, 3375, 3380, 3390 or LGDK.

When using a 3350, specify the drive format used—3330 or 3350.

LGDK is a generic definition for disk drives that have a track size greater than 32767 bytes. The 3375, 3380, 3390, and future devices can also be specified as LGDK.

The second parameter specifies the buffer size to be used for the data set. The minimum buffer size is 1024. The maximum allowable size is the track size for the device type specified or 32767, whichever is smaller. The default is 4096.

The third parameter is no longer used. If specified, it is ignored.

SVCNO=

Specifies SVC numbers reserved for use by the generated system.

The first parameter specifies the type 2 SVC number. The entered value can range from 200 to 255. The default is 254. The type 2 SVC number is required for batch and online systems. One type 2 SVC number can be used for any number of IMS systems of the same release level.

The second parameter specifies the type 4 SVC number reserved for use by database recovery control (DBRC). The entered value can range from 200 to 255. The default is 255. One type 4 SVC number can be used for any number of IMS systems of the same release level.

For reasons of compatibility, the SVC numbers, if specified, must be immediately preceded by a comma and then enclosed in parentheses. Also, the type 2 and type 4 SVC number cannot be the same.

IMSCTF Macro

Example: Following is an example of the IMSCTF macro instruction for the following condition:

- Type 2 SVC is 254.
- DBRC Type 4 SVC is 255.
- Checkpoint frequency is every 2000 log records.
- The maximum dynamic storage allowed is 16K.
- The increment is 4K.

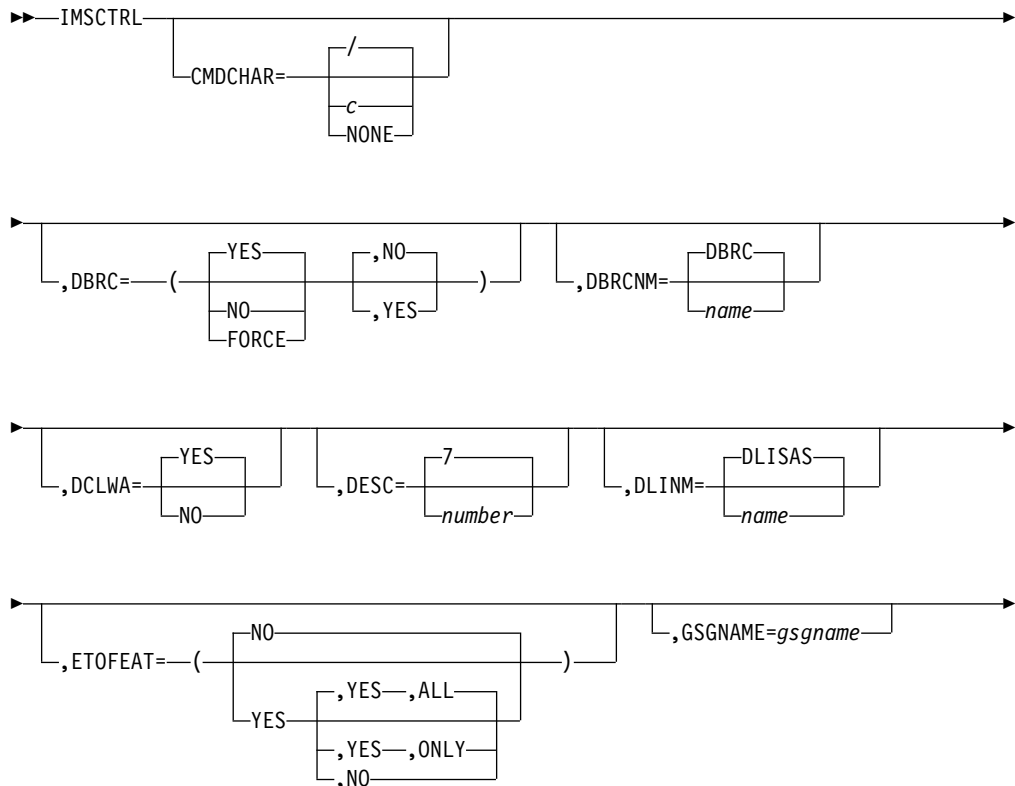
```
IMSCTF SVCNO=(,254,255),APNDG=(,Z0),CPL0G=2000
```

IMSCTRL Macro

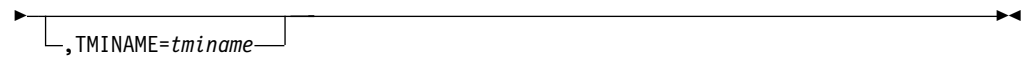
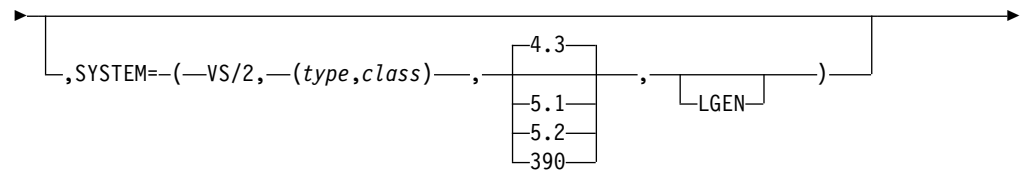
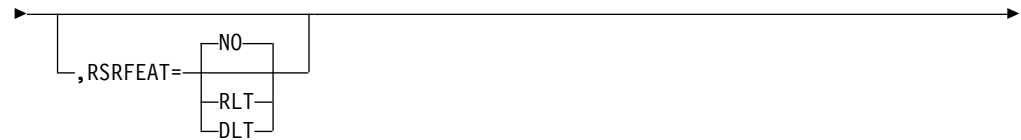
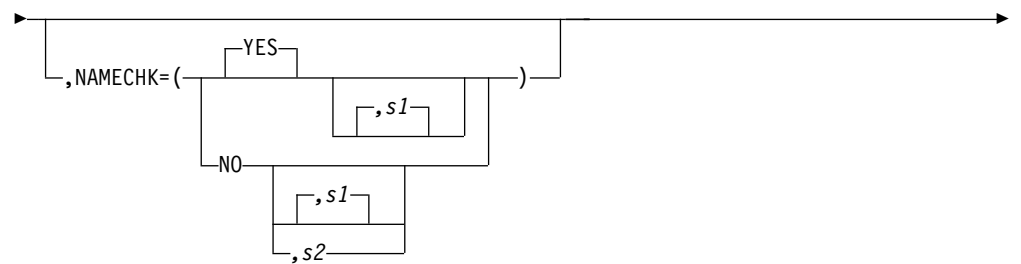
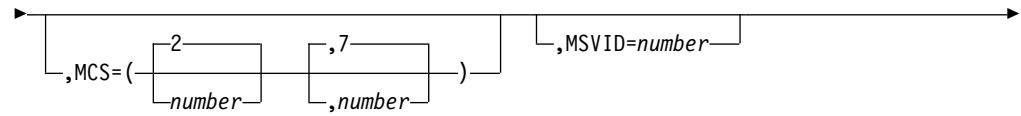
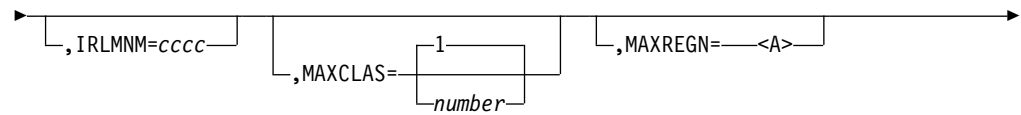
The IMSCTRL macro statement describes the basic IMS control program options, the MVS system configuration under which IMS is to execute, and the type of IMS system definition to be performed.

IMSCTRL is required. This macro instruction must be the first statement of the system definition control statements that are entered into the system definition Stage 1 for an IMS or a DBCTL system. You can have only one IMSCTRL macro instruction within an IMS system definition.

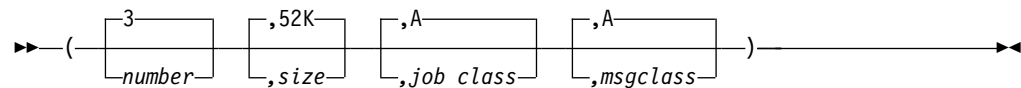
The only other macro instruction required for a batch environment is IMSGEN. The IMSCTF macro instruction is optional.



IMSCTRL Macro



<A>



IMSCTRL Macro

Keyword Parameters

To find which parameters apply to your IMS configuration, refer to Table 12 on page 32.

CMDCHAR=

- /** The default value.
- c** Any character other than comma (,). Ampersand must be entered as && to define & as the command character.

NONE No command recognition character (CRC) is defined.

CMDCHAR can be specified for DBCTL, DB/DC or DCCTL. The value of CMDCHAR can be specified at execution time by specifying execution parameter CRC=c.

With a DBCTL system, CMDCHAR specifies the command recognition character used by the MTO to enter IMS commands for this DBCTL environment. If CMDCHAR=NONE is specified and the CRC is not specified as an execution parameter (CRC=), then commands are entered for this DBCTL system with a prefix of the IMSID.

DB/DC and DCCTL systems allow commands to be entered at MCS/E-MCS consoles with a CRC prefix by specifying execution parameter keyword CDMMCS with a value other than N.

The CRC can be specified with CMDCHAR or with CRC at execution time.

If CMDCHAR= is specified and the CRC is not specified as an execution parameter (CRC=), commands are entered for the DB/DC or DCCTL system with a prefix of the IMSID.

Although other products and subsystems might use certain command recognition characters, the CRC is not required to be unique among all IMS systems. Choose an unreserved character that is appropriate for your particular system.

Do not use a character for the CRC that is the same as the beginning character of an MVS command. If the CRC is the same as the beginning character of an MVS command, that command does not work after you start IMS. For example, if you start IMS with CRC=D, MVS does not respond to any of the MVS display commands such as D A,L.

The CRC value you select might affect multisegment command processing. The last character of the command is checked during multisegment command processing. If that character matches the CRC, the command is assumed to be a multisegment command. Processing of the command waits for the rest of the command. For example, if you use B for the command recognition character, and issue the command 'BMODIFY PREPARE ACBLIB', the command does not run. The ending 'B' in 'ACBLIB' is interpreted to indicate a multisegment command. To enter this command correctly, enter:

```
'BMODIFY PREPARE ACBLIB . '
```

The character you choose for a DB/DC system becomes the command recognition for the IMS system. If your installation allows commands from MCS or E-MCS consoles, you need a CRC only if both of the following statements are true:

- IMS provides DBCTL service to a CCTL.
- CCTL broadcasts an IMS SWITCH command preceded by a CRC.

CMDCHAR does not affect the normal DB/DC MTO's use of the / character when entering IMS commands.

DBRC=

Specifies whether DBRC is to be active in the online and batch environments. You specify two DBRC= parameters. The first parameter specifies the online system and the second parameter specifies the batch system.

In the first parameter, DBRC=YES and DBRC=NO are no longer used, because DBRC is always active in the online configuration. However, they are retained for compatibility, and if specified for the online configuration, they are ignored. Their specification still affects a batch execution, as explained later.

If FORCE is specified for the first parameter, the second parameter **must not** be specified. Specifying DBRC=FORCE for the first parameter means that DBRC is to be used in **both** the online and batch configurations.

When changing this parameter to FORCE, or from FORCE to another option, either a BATCH or an ALL system definition is required to make the change effective in the batch environment. This specification **cannot** be overridden by specification of the DBRC= EXEC parameter for batch or in the EXEC statement of applicable IMS utility programs, with the following exceptions:

- A batch backout execution of IMS
- Execution of the System Log Recovery utility
- Execution of the IMS Log Archive utility

That is, if FORCE is specified here, it applies to all executions of IMS except those listed above.

DBRC=FORCE is recommended for all IMS subsystems in an RSR complex.

The second parameter specifies whether (YES) or not (NO) DBRC will be used in the batch environment. Either of these specifications can be overridden by specifying the DBRC= EXEC parameter on the DBBBATCH or DLIBATCH procedures. The default is NO, indicating that DBRC is not to be used in the batch environment.

The second parameter specifies the default use of DBRC by IMS database utilities that interface to DBRC.

Related Reading: For more information, see *IMS/ESA Utilities Reference: Database Manager*.

If the second DBRC= parameter is omitted, DBRC is not active in the batch environment unless YES or FORCE is specified for the first parameter. That is:

- If neither the first nor the second parameter is specified, DBRC is active in the online but not in the batch environment.

IMSCTRL Macro

- If the first parameter is specified or the default of NO is accepted and the second of parameter is omitted, DBRC is active in the online but not in the batch environment.
- If the first parameter is specified as YES and the second parameter is omitted, DBRC is active in both environments unless overridden in the batch environment with an EXEC parameter.
- If the first parameter is specified as FORCE, DBRC is active in both environments, and its specification cannot be overridden.

DBRCNM=

Specifies for MVS the IMS.PROCLIB member containing the skeletal procedure for the DBRC address space, produced by IMS system generation. The IMS control region automatically initiates the DBRC address space, using the MVS start command and specifying this name. To rebuild the DBRC procedure, you must use the IMS.PROCLIB member containing the skeletal procedure for the DBRC address space produced during system generation. This procedure contains the name you specify in stage 1. You must first change DFSPBIMS, DFSPBDBC, or DFSPBDCC to reflect the specified DBRCNM, and then perform a regen.

Consequently, the DBRC procedure produced by IMS system generation must be copied from IMS.PROCLIB to SYS1.PROCLIB. For more information, see “Making IMS and IMSRDR Procedures Accessible to MVS” on page 382.

You can specify up to an eight-character name, the first character being alphabetic. The default name is DBRC.

Specifying DBRCNM=xxxxxxx results in the following:

A DBRC address space cataloged procedure is created with the name xxxxxxxx.

To make the DBRC address space cataloged procedure name of xxxxxxxx operational, the following must be true:

A DBRC address space must be cataloged as xxxxxxxx in SYS1.PROCLIB.

Simply specifying DBRCNM=xxxxxxx in your IMS system definition is not sufficient.

DCLWA=

Establishes a default value that IMS applies to any transactions defined by subsequent TRANSACT macro statements. DCLWA= specifies whether (YES) or not (NO) IMS should perform log write-ahead for recoverable nonresponse-mode input messages and transaction output messages. The default is YES. If no value is specified for DCLWA= on the IMSCTRL macro, the value for DCLWA= on any subsequent TRANSACT macros accepts the default of YES, or you can specify YES or NO on the individual TRANSACT macros.

Specify or accept the default of YES to ensure that:

- A nonresponse–input transaction is made recoverable across IMS failures, prior to IMS acknowledging receipt of the input.
- Database changes are made recoverable prior to IMS sending associated output reply messages.

YES ensures that information in the log buffers is written to the IMS log before the associated input acknowledgement or output reply is sent to the terminal.

Specify or accept the default of YES for all VTAM terminal types. For BTAM terminals, selection of this option can increase input and output message path lengths.

Related Reading: For information about performance implications of selecting log write-ahead for transactions, see *IMS/ESA Administration Guide: System*.

Specify NO if input message integrity and the consistency of output messages with associated database updates is not required. DCLWA does not apply to response mode or Fast Path input processing, and, if specified in these situations, is ignored during IMS execution.

DESC=

Specifies the message descriptor code to be assigned to the IMS system console messages if MCS support is included in the MVS generation. If DESC is not specified, the default is 7.

The IMS MCS= and DESC= keywords should be defined as is required for the ROUTCDE and DESC keywords of the MVS WTO macro.

DLINM=

Specifies for MVS the IMS.PROCLIB member to contain the skeletal procedure for the optional DL/I address space produced by IMS system generation. The IMS control region automatically initiates the DL/I address space using the MVS start command and specifying this name.

Consequently, the DL/I procedure produced by IMS system generation must be copied from IMS.PROCLIB to SYS1.PROCLIB. You probably need to modify the skeletal procedure.

For more information, refer to “Member Name DLISAS” in “Chapter 4. Procedures” on page 189.

You can specify up to an eight-character name, the first character being alphabetic. The default name is DLISAS.

Specifying DLINM=xxxxxxx results in the following:

A DL/I address space cataloged procedure is created with the name
xxxxxxx.

ETOFEAT=

Specifies whether or not the ETO feature is to be included in your IMS system. The default is NO.

The ETO feature allows you to dynamically create and delete ACF/VTAM terminals, users, and MSC RCNTS based on model descriptor statements rather than a static IMS system generation.

If you have installed the ETO feature and it is available within the libraries after stage 1 and stage 2 system definition, the first parameter indicates whether (YES) or not (NO) to include ETO in your IMS system. If you have not installed ETO, the ETO option is forced off.

If ETO is included, the second parameter indicates whether (YES) or not (NO) descriptors should be produced. If this parameter is null, the default value depends on how the first parameter is specified. If it is YES, a null second parameter uses the default of YES. If the first parameter is NO, a null second

IMSCTRL Macro

parameter is also NO (if the first parameter is specified or uses the default of NO, do not specify the second parameter).

The third parameter indicates whether (ALL) or not (ONLY) system definition should only produce descriptors, or if it should produce descriptors along with a normal system definition. If the second parameter is specified or uses the default of YES, ALL is the default. If the second parameter is specified or uses the default of NO, do not specify the third parameter.

GSGNAME=

Specifies the global service group (GSG) name to be used for the RSR complex. No default value exists.

If you do not specify the GSG name in the IMSCTRL macro, it must be specified in the DFSRSRxx member (or the JCL for batch or utility jobs) to enable RSR. If you do specify the GSG name in the IMSCTRL macro, you might have to change batch JCL to allow batch jobs to run with RSR not enabled.

HSB=

Specifies whether (YES) or not (NO) an XRF-capable system is to be generated. The default is NO.

Specifying NO prevents the possibility of an XRF capability without rerunning the system definition and an IMS cold start.

It is recommended that you code HSB=YES on the IMSCTRL macro if you are planning on running XRF. Having HSB=YES coded might eliminate a subsequent IMS system generation. XRF is disabled if no value is specified for HSBID in the DFSPBIMS or DFSPBDCC members, or in the IMS execution JCL. Informational message DFS3899I is issued.

IMSID=

Specifies a one- to four-character alphanumeric identifier for the IMS system.

The IMSID value specified is used as the IMS subsystem identifier. For online control regions, it must be different from any other IMS subsystem identifier or non-IMS subsystem identifier defined to the operating system under which IMS is running. This identifier is also used to relate messages that are routed to the MVS system console with the corresponding IMS system. To avoid confusion as to which region has issued a console message, it is recommended (though not required) that the IMSID be unique for batch regions. The default is IMSA.

If you have renamed the IMS procedure (the procedure that starts IMS), the IMSID value specified must not be the same as the IMS procedure name unless one of these conditions exists:

- All the DD statements in the startup procedure are cataloged in the master catalog
- The unit and volume are specified on each DD statement

IRLM=

If coded as IRLM=NO, sets the IRLM batch default to NO. This lets you generate IMS with IRLM active for the online system, but inactive for the batch environment. The NO can be overridden by the IRLM=Y parameter in the batch JCL.

If the IRLM= parameter is omitted, and the IRLM name is specified, the IRLM batch default is set to YES, but can be overridden by the IRLM=N parameter in the batch JCL.

If IRLMNM is not coded, this parameter is ignored. (See the IRLMNM parameter.)

To avoid assembly errors, including 2,G006, you must specify the combination of the IRLM and IRLMNM parameters in one of the following ways:

- If you specify IRLM:
 - IRLM=Y and IRLMNM=xxxx
 - omit IRLM and specify IRLMNM=xxx
- If you omit IRLM:
 - IRLM=N and IRLMNM=xxxx.
 - omit **both** parameters

If you specify the IRLM= parameter and omit the IRLMNM= parameter, assembly errors can result.

IRLMNM=

Specifies a one- to four-character alphanumeric name assigned to the Internal Resource Lock Manager (IRLM) included in your system.

If the IRLMNM= parameter is not specified, the IRLM is not included in your IMS system. To activate the IRLM, you must code IRLM=Y in the execution JCL.

When the IRLMNM= is not specified in the IMSCTRL macro or in the execution JCL, and IRLM=Y is specified in the execution JCL, the IRLM name used is IRLM.

The IRLM must be active in your system if your IMS system is to share data at the block level with other IMS systems. The IRLM is optional for systems that do not share data at the block level.

Only one IRLM can be specified for any IMS system (and thus, only one IRLMNM), but multiple IRLMs can exist within a particular MVS system. Each IRLM must be defined as an MVS subsystem.

Related Reading: For more information, refer to *IMS/ESA Installation Volume 1: Installation and Verification*.

The name specified on this IRLMNM= parameter must be the MVS subsystem name assigned to the IRLM with which this IMS system is to connect. If IMS is connected to an executing copy of IRLM, DB2 cannot connect to it. If DB2 is connected to an executing copy of IRLM, IMS cannot connect to it.

If this parameter is specified and not overridden by means of the IRLM=NO parameter on the online or batch startup procedures, IMS only completes initialization if the named IRLM is available at execution time. Also, IMS uses the named IRLM for all locking services, both local and sharing. Program isolation locking is not used.

IMSCTRL Macro

It is important to understand the interaction of this IRLMNM parameter with the DBRC parameter and with the ACCESS parameter on the DATABASE macro statement.

Related Reading: For additional information on administering a data-sharing environment, see *IMS/ESA Administration Guide: System* before defining your data sharing options.

MAXCLAS=

Specifies the range of active (started) classes after IMS initialization. Valid values are decimal numbers from 1 to 255. The default is 1. Any transaction code class specified on either the APPLCTN or TRANSACT macro statement must not exceed this value. If the class is not active at the time of scheduling, the schedule request is treated as if the class were stopped.

MAXIO=

Is no longer used. If MAXIO is specified the value is ignored.

MAXREGN=

Specifies the maximum number of IMS control block sets that are permanently allocated. A control block set is required for each dependent region type (batch message processing, message processing, or CCTL threads) that can be started. (Note that this does not imply that the same number of dependent regions are always active.)

This value must be a decimal number from 1 through 999. When the count of dependent regions started exceeds the value specified by the MAXREGN= parameter, control block sets are dynamically allocated, but the storage for these sets is released as the dependent regions terminate. The default is 3.

For an RSR tracking subsystem, the value specified on the MAXREGN= parameter determines the maximum number of PSTs that are to be used for DL/I database tracking. This number should be twice the number of DL/I PSTs used in all of the active subsystems tracked by the tracking subsystem. If many PSTs are non-updating, set this number to twice the number of updating DL/I PSTs used in all active subsystems. A minimum of two PSTs is used.

The second through fourth fields specify region size (default value is 52K), region job class (default class is A), and job message class (default class is A).

Region size (which must be expressed in terms of xxK), region job class, and job message class must conform to the operating system JCL specifications. This operand is ignored for BATCH system definitions.

MCS=

Specifies the MVS routing code to be assigned to the IMS system console if multiple console support (MCS) is included in the operating system. If MCS is not specified, the default is (2,7).

Although MVS supports more than 16 route codes, IMS uses only route codes 1 through 16.

In a DBCTL environment, MSC= defines which consoles are to receive unsolicited DBCTL messages.

MSVID=

Specifies a one- to three-digit decimal number between 1 and 676 that is used to complete the eight-character name of the generated IMS Multiple Systems

Verification utility control blocks module generated for this system (DFSMSxxx). If fewer than three decimal digits are specified, the value is right-justified and filled to the left with zeros. (For example, specifying 3 creates the name DFSMS003.)

This control blocks module is assembled and link-edited during IMS system definition stage 2 processing. The Multiple Systems Verification utility procedure (IMSMSV) is, as the user directs, placed into the IMS.PROCLIB data set. This keyword is required when MSVERIFY is specified in the SYSTEM keyword.

This keyword is ignored if no multiple systems definition statements are present. The keyword must be present to generate the IMS Multiple Systems Verification utility control blocks and procedure.

This keyword is ignored for DBCTL.

The specification of MSVID on the IMSCTRL macro statement is not related to the specification of SYSID on the MSNAME macro statement.

NAMECHK=

Specifies whether (YES) or not (NO) resource name checking and cross-checking are to be performed, and, if NAMECHK=NO, whether sorting is to be performed in stage 1 (S1) or stage 2 (S2).

If duplicate resource name checking and resource name cross-checking will be performed (that is, NAMECHK is specified as or uses the default of YES), sorting must be performed in stage 1 (S1 specified or used by default). If resource name checking is to be bypassed, sorting can occur in either stage 1 (S1 specified or by default) or stage 2 (S2 specified).

It is suggested that you use the default values for NAMECHK unless:

- You have run the preprocessor prior to your stage 1 run and you want to omit resource name checking here (by specifying NAMECHK=NO).
- You have a specific performance reason for having IMS sort in stage 2, (for example, you generally run stage 1 multiple times per stage 2).

Related Reading: For information about the options available to you by including or omitting the preprocessor in combination with options on the NAMECHK parameter, see "Verifying the Stage 1 Input" in *IMS/ESA Administration Guide: System*.

This keyword is ignored if LGEN is specified in the SYSTEM keyword.

RSRFEAT=

Specifies whether (RLT or DLT) or not (NO) an RSR-capable system is to be generated. The default is NO.

RLT is a minimum RSR-capable system. IMS tracks recovery information for all covered databases but does not update them at the tracking site.

For DLT, IMS tracks recovery information for all covered databases and updates them at the tracking site.

Specifying NO prevents the possibility of an RSR capability without rerunning the system definition and an IMS cold start.

IMSCTRL Macro

Specifying RLT or DLT and not having the appropriate RSR feature present in your stage 1 assemblies causes system definition to fail. A message G1205 is issued.

Specifying the DLT feature requires that both the RLT and DLT features be present in your stage 1 assemblies. Specifying RLT requires only the RLT feature to be present in your stage 1 assemblies.

DLT cannot be specified for a DCCTL class sysgen.

SYSTEM=

Specifies the operating system configuration and the type of IMS system definition to be performed.

The first SYSTEM= parameter must be VS/2, indicating the operating system is MVS.

The second SYSTEM= parameter consists of two subparameters. The first subparameter defines *type* of system definition. The definition types are: ALL, BATCH, CTLBLKS, MODBLKS, MSVERIFY, NUCLEUS, and ON-LINE.

When you first install IMS, you must perform a full system generation. Thereafter, you can perform a subset of the full system generation to add or change a function. The parameters used in these subset system generations are described below. The type of generation required to add or change a function is described in "Selecting the Appropriate IMS System Definition" on page 29.

The SMP JCLIN process should be run after the stage 1 of all generation types.

After an ALL, ON-LINE, NUCLEUS, CTLBLKS, or MODBLKS generation is completed, the Security Maintenance utility must be rerun.

ALL

Combines the BATCH and ON-LINE options.

BATCH

Moves required modules from the IMS distribution libraries to the user's libraries; it also generates system procedures and a database system.

A BATCH generation should not be run to add maintenance or to add or delete database functions or features to a residence library that contains a nucleus with the Data Communication feature.

CTLBLKS

Generates control block modules of all IMS control blocks for use within an IMS nucleus in the IMS control program region. The blocks built in the MODBLKS option are also included in this generation. This subparameter should be specified to replace the control blocks of an existing nucleus. If you are performing a CTLBLKS generation, the new nucleus must have the same suffix as the old nucleus. SMP maintenance integrity is lost if you rename the nucleus prior to using it in a CTLBLKS generation because SMP does not recognize the contents of the renamed nucleus. To change the control blocks in an existing nucleus, do a CTLBLKS generation. To generate a new nucleus, do a NUCLEUS generation.

The SMP JCLIN process should be run after the stage 1 of a CTLBLKS generation.

A CTLBLKS specification generates the MSC Verification Utility control blocks, if MSC definitions are present. When you specify CTLBLKS, you need not perform a separate system definition with MSVERIFY.

If any device or option is to be deleted during a CTLBLKS generation, unresolved references may exist when the nucleus is link-edited. This can also happen if logical terminal names are added or deleted before the occurrence of master or secondary master logical terminal names.

MODBLKS

Generates the control blocks members for resources to be added or changed online. These control blocks are used by the IMS control region, the Security Maintenance utility (SMU), and the Multiple Systems Coupling Verification utility.

The MODBLKS system definition requires that the stage 1 input contain all source statements for the existing system to which the online changes are to be made, with the exception of communication specifications. If you use the Multiple Systems Coupling MSVERIFY utility, **all** source statements, including communication specifications, are required input to your stage 1 run. However, if communication specifications are omitted from a MODBLKS system definition, the following stage 1 error checking is not performed:

- Transaction names are not cross-checked against LTERM names to locate duplicates.
- For MSC users, the remote system ID (SYSID) on the TRANSACT macro statement is not checked against the remote system ID (SYSID) on the MSNAME macro statement to ensure that they are not duplicates.

Thus, omitting communication statements can result in undetected system definition errors. You must therefore consider your requirements for improved stage 1 performance against the possibility of undetected errors in your stage 1 input to determine whether you want to include or omit communication statements from your MODBLKS system definition input.

Only APPLCTN, DATABASE, TRANSACT, and RTCODE statements can be added, deleted, or modified. The restrictions pertaining to these modifications are found in the descriptions of these macro statements. IMS does not enforce these restrictions; rather, you are responsible for adhering to the directions provided there.

The effects of a MODBLKS system definition are made known to the IMS online system by means of a master terminal operator /MODIFY command sequence.

Stage 2 assembler output for a MODBLKS system definition is placed in IMS.OBJDSET. Linkage editor output for the control blocks members affected by running an online change is placed in IMS.MODBLKS.

Certain types of changes, when made across a MODBLKS system definition, do not affect an existing resource except across a cold start. That is, the effects of prior commands are not negated by online change procedures. For example, resources that have been modified by commands such as /START, /STOP, /ASSIGN, /MSASSIGN are not altered, nor are

IMSCTRL Macro

resources whose status has been altered because of changes in the operating configuration (for example, a resource stopped because of an error condition).

MSVERIFY

Generates a subset of the online control blocks required for execution in a multiple-IMS system configuration. MSVERIFY also generates the link-edit JCL and control statements required to place the Multiple Systems Verification utility into the IMS.RESLIB data set. MSVERIFY can also generate the JCL and control statements to place the IMSMSV procedure into the IMS.PROCLIB data set. This subparameter should be specified to generate a test system to verify, prior to attempting online execution, the consistency of definitions in an MSC configuration. This keyword causes an assembly error if the multiple systems definition statements are not present. If MSVERIFY is specified, MSVID is required.

NUCLEUS

Generates an IMS nucleus for the IMS control program region and control block modules for all IMS control blocks. This subparameter should be specified when generating an alternate nucleus that includes new or deleted old system features, such as additional terminal support or conversational support. A NUCLEUS specification generates the MSC Verification utility control blocks if MSC definitions are present. If NUCLEUS is specified, you do not need to perform a separate system definition, with MSVERIFY specified, to obtain the blocks.

To change the control blocks in an existing nucleus, do a CTLBLKS generation. To generate a new nucleus, do a NUCLEUS generation.

The SMP JCLIN process should be run after the stage 1 of all generation types.

ON-LINE

Moves required modules from the IMS distribution libraries to the user's libraries, generates system procedures, and generates those modules that contain the CTLBLKS and NUCLEUS options for an IMS system. It generates the assembler language and link-edit statements for all operating systems and VTAM-sensitive modules that are supplied in source form.

Requirement: You must specify ON-LINE with the hyphen.

Guidelines for Selecting the System Definition Type

Guidelines are provided to assist you in selecting the CTLBLKS, NUCLEUS, ON-LINE, or MODBLKS system definition type to use when modifying an existing ALL type of system definition. The guidelines should help you in determining whether modification of parameters at execution can preclude the need for a new system definition. For more information, see "Selecting the Appropriate IMS System Definition" on page 29.

Example: The following is an example of the IMSCTRL macro with:

- Three message regions
- Message region job class A (default)
- Job message class C
- Online and batch

```
IMSCTRL SYSTEM=(VS/2,(ALL,DB/DC),4.3),X  
MAXREGN=(3,52K,,C)
```

The second SYSTEM= subparameter defines the system definition environment class. The three system definition environment classes are:

DB/DC	Builds a standards IMS system
DBCTL	Builds only the DBCTL environment
DCCTL	DCCTL class builds only the DCCTL environment

DBCTL, DCCTL, and DB/DC (with or without XRF) systems are mutually exclusive and cannot share the same IMS.RESLIB. DB/DC is the default. See the description of the SUFFIX= parameter under the IMSGEN macro for additional restrictions concerning the use of one IMS.RESLIB to support multiple systems.

If you do not specify DBCTL on the second subparameter, the following are always included:

- All enhanced security codes and commands
- All MFS code, commands, procedures, and default formats
- VTAM code and commands, whenever VTAM is requested

The MFS format library is a required data set (regardless of whether MFS is used), except for DBCTL systems.

The MSVERIFY,DBCTL type/class combination is invalid. An error message is generated for invalid combinations. DBCTL and DB/DC (with or without XRF) systems are mutually exclusive and cannot share the same IMS.RESLIB.

The third SYSTEM= parameter indicates the release and modification level of the operating system; the default is 4.3 (MVS/ESA System Product Version 4 Release 3).

If you are using the OS/390 operating system specify the third SYSTEM= parameter as 390. Use 390 to specify all releases of OS/390.

The fourth SYSTEM= parameter defines the LGEN configuration. Use LGEN if a stage 1 or any of the stage 2 job steps of a system definition requires more than 4 MB of private storage for execution.

The LGEN parameter is only valid when stage 1 source statements are input to the preprocessor. The preprocessor passes a group of up to 10000 resources to the assembler for stage 1 processing. The assembler is invoked as many times as is necessary to process all appropriate resources.

Related Reading: For more information on the LGEN configuration and how the IMS control region is organized, refer to *IMS/ESA Administration Guide: System*.

TMINAME=

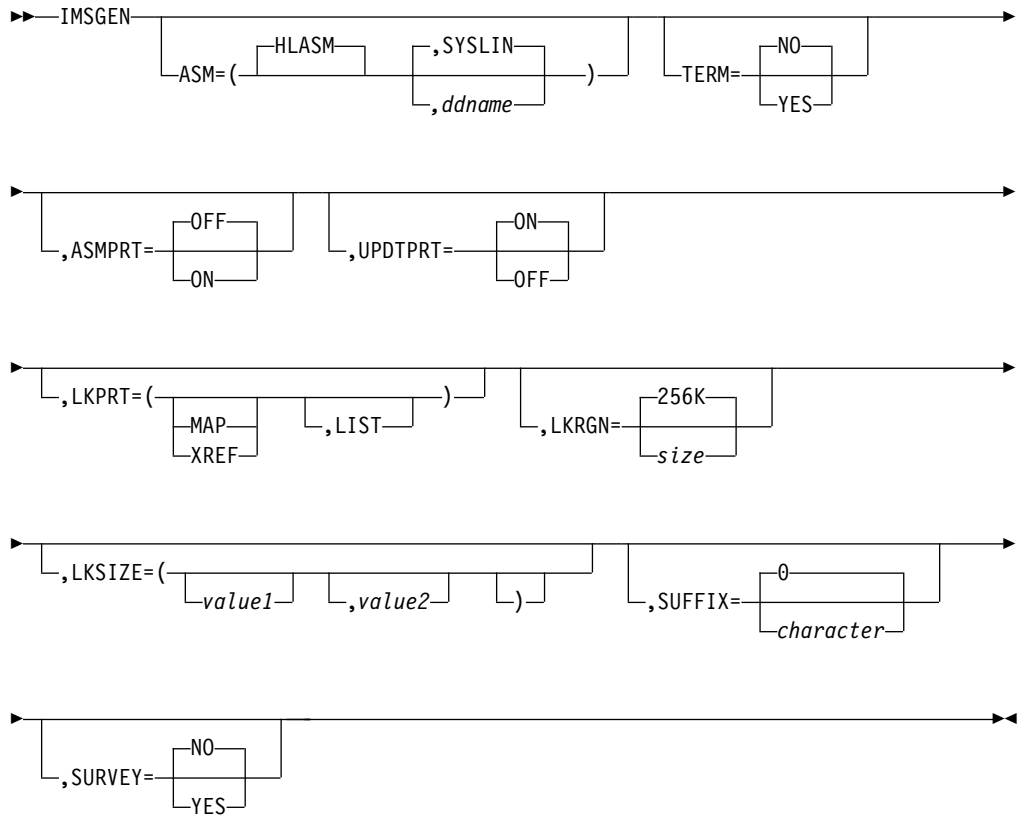
Specifies the one- to four-character name that becomes the instance name for the transport manager subsystem (TMS) and the subsystem name (SSN) for an RSR active or tracking subsystem. If TMINAME= is not specified, IMS uses a series of blanks as the default, which associates IMS with the TMS that used a default instance name.

IMSGEN Macro

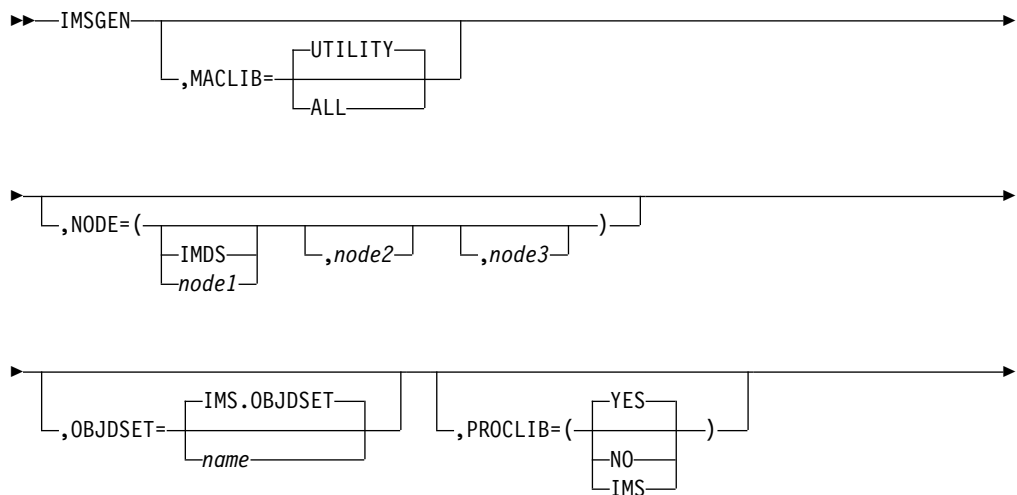
IMSGEN specifies the assembler and linkage editor data sets and options, and the system definition output options and features.

IMSGEN Macro

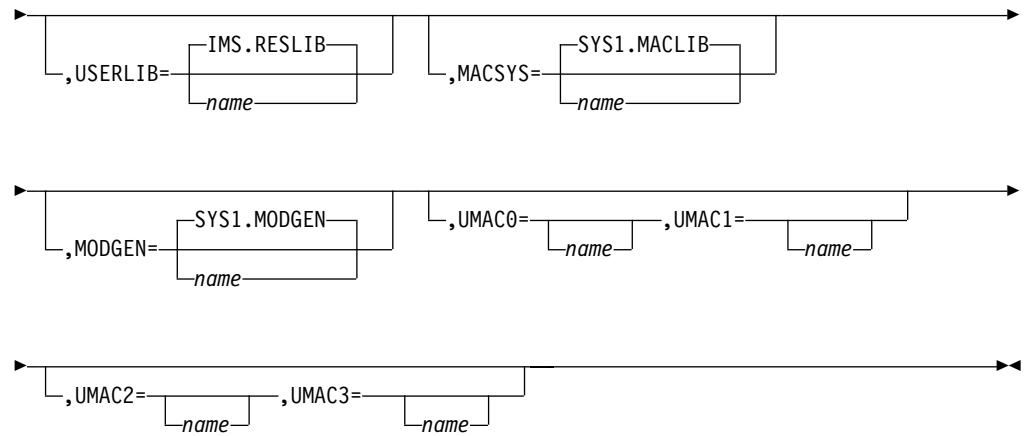
IMSGEN must be the last IMS system definition macro in the stage 1 input stream, and it must be followed by an assembler END statement.



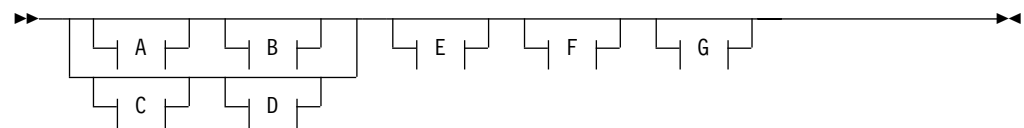
IMS Data Set Options



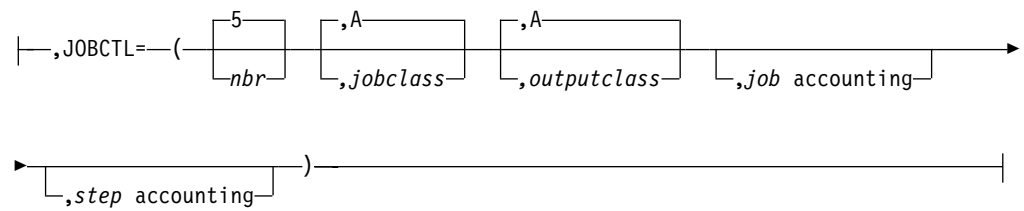
IMSGEN Macro



Job Control Language Options



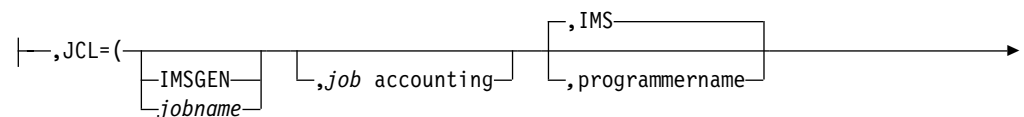
A:



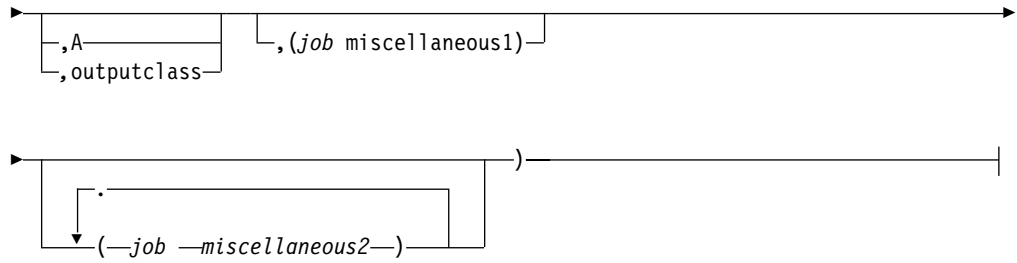
B:



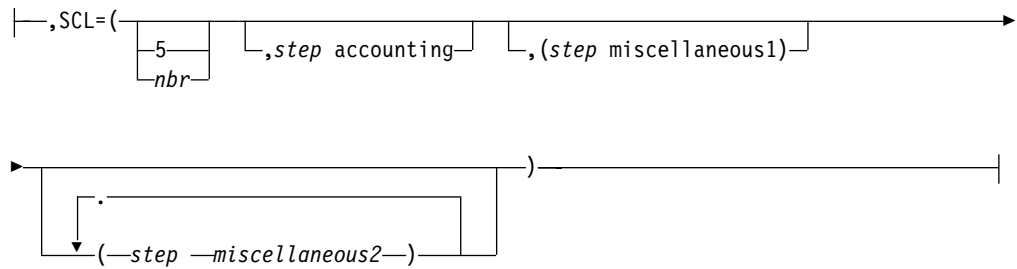
C:



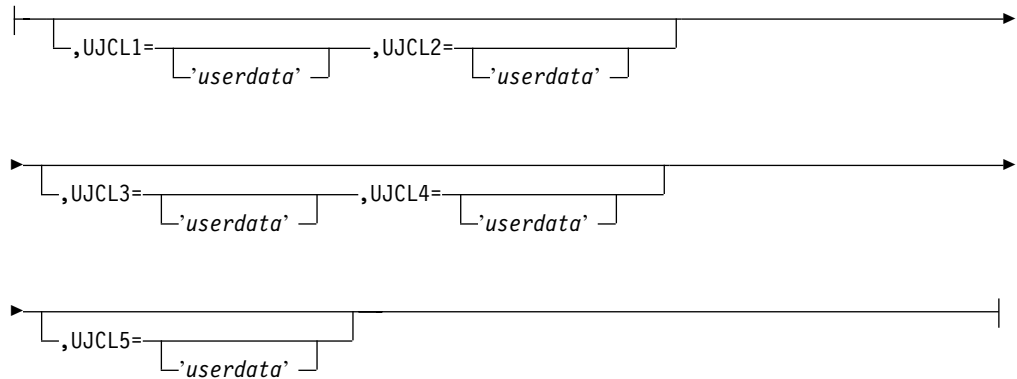
IMSGEN Macro



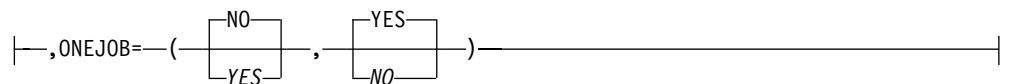
D:



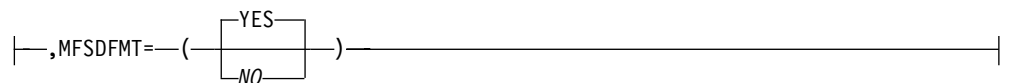
E:



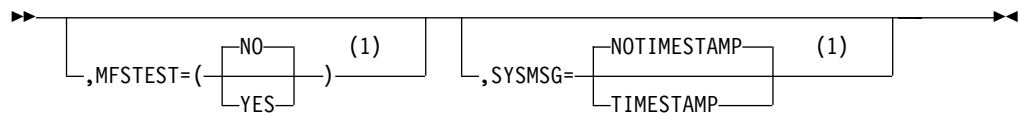
F:



G:



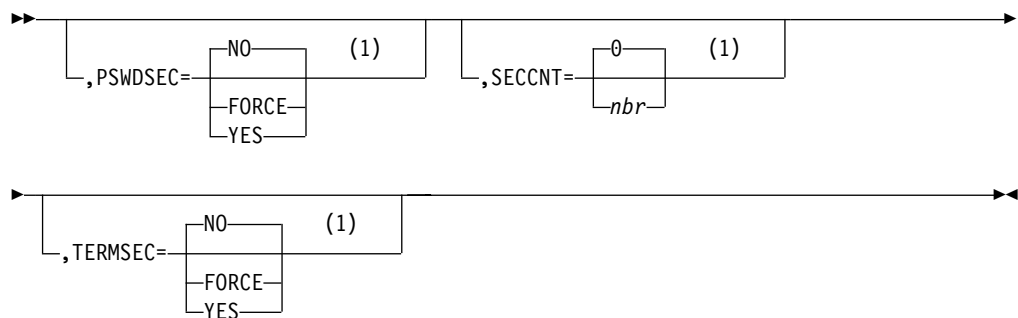
General Communication Options



Notes:

- 1 If you specify the COMM macro, the general communication option parameters are overridden.

Security Options



Notes:

- 1 If you specify the COMM or SECURITY macro, the security option parameters are overridden.

Keyword Parameters

To find which parameters apply to your IMS configuration, refer to Table 12 on page 32.

Assembler and Linkage Editor Options

ASM=

The first parameter specifies the assembler JCL to be produced for the stage 2 assembly steps. HLASM, the high level assembler, is the only valid first parameter.

For an LGEN assembly, the high level assembler is used in the stage one assembly cycles.

The second parameter can be used to specify the ddname to be used for the assembler object output. If the ddname is not specified, the ddname is SYSLIN.

Avoid the use of ddnames other than SYSLIN, SYSGO, and SYSPUNCH if you are using the MVS System Modification Program (SMP) to apply maintenance to your IMS libraries. SMP does not recognize other ddnames during JCLIN processing.

The option of changing the object output ddname is intended to accommodate those users who have installed an assembler and altered its ddname defaults. The stage two assembler execution statement options are PARM='OBJECT,NODECK,NODBCS'. The object file output ddname is the one

IMSGEN Macro

that must be coded in place of the default value of SYSLIN. Also, you must use the default assembler option ALIGN for assemblies of IMS modules.

ASMPRT=

Specifies the assembler print options for those assembler job steps produced by IMS system definition.

ON specifies that assembly listings are to be generated.

OFF specifies that assembly listings for inline assemblies are not to be generated. The default is OFF.

LKPRT=

Specifies linkage editor print options for those linkage editor job steps produced by IMS system definition. The print options are described below:

Value Print Option

MAP Module map

XREF Cross-reference table (XREF includes the MAP option)

LIST List of control statements in statement image format

If this parameter is omitted, only linkage editor error messages, if any, are printed.

LKRGN=

Specifies a region size value to be placed on the generated EXEC statements for execution of the linkage editor. This parameter can be specified as a decimal number (not to exceed 999999) or in the form *nK* (not to exceed 99999K). The default is 256K.

LKSIZE=

Specifies the values to be placed in the SIZE parameter of the EXEC statements for the VS link-editing jobs.

value1

Specifies the maximum number of bytes of main storage available to the linkage editor. This value can be specified in the form *n* (where *n* represents the actual number of bytes of main storage, not to exceed 99999) or *nK* (where *n* represents the number of 1K blocks of main storage, not to exceed 9999K).

value2

Specifies the maximum amount of *value1* that is to be used as the load module buffer. It is expressed in the form *n* (where *n* cannot exceed 9999) or *nK* (which should not exceed 100K).

Default values are not supplied in LKSIZE for the link-edit SIZE parameter. If *value1* and *value2* are omitted, the link-edit SIZE parameter defaults to specifications chosen during VS system generation, if any.

Related Reading: For a more detailed description of the SIZE option, see *MVS/ESA Linkage Editor and Loader User's Guide*.

SUFFIX=

Specifies the alphanumeric suffix character appended to the generated composite control block, nucleus, MFS device characteristics table, and security directory module names when they are put into the IMS.RESLIB or IMS.MODBLKS data sets. The IMS online nucleus name always starts with

DFSVNUC. The suffix character supplies the eighth character of the nucleus names. If the suffix character equals 0, the online nucleus name is DFSVNUC0. This concept allows the system user to generate multiple IMS systems for use in one environment where the characteristics of each system vary. The default is 0.

All systems generated from the IMS.RESLIB must use the same functions and features. These systems include:

- MSC
- Fast Path
- DB/DC, DBCTL, or DCCTL
- ETO
- DBRC
- IRLM

The modules that receive the suffix are:

DFSBLK0x	DFSSCD0x, DFSSCD1x
DFSCLCxx	DFSFRB0x, DFSFRB1x
DFSCLIDx	DBFSCD0x, DBFSCD1x
DFSCLRxx	DFSPDIRx
DFSCLSxx	DFSRCTEx
DFSCLVxx	DFSSMB0x
DFSDDIRx	DFSUDT0x
DFSOPLOx	DFSVNUCx
DFSISDBx	
DFSISDCx	

The suffix is also appended to the intermediate control block assemblies for SMP/E purposes.

SURVEY=

Specifies whether (YES) or not (NO) the Database Surveyor Utility feature is to be included in the system. The default is NO.

TERM=

Specifies the assembler TERM/NOTERM option for stage 2 assembler job steps produced by the IMS system generation.

YES specifies that the TERM option is in the PARM= field for each of the stage 2 assembler job steps. YES also specifies that the //SYSTEM DD statement is to be built in each of the stage 2 assembler job steps.

NO specifies that the NOTERM option is included in the PARM= for each of the stage 2 assembler job steps. The default is NO.

UPDTPRT=

Specifies whether the IEBUPDTE SYSPRINT is to be produced by IMS system definition stage 2.

ON specifies that the IEBUPDTE SYSPRINT is to be generated. The default is ON.

IMSGEN Macro

OFF specifies that the IEBUPDTE SYSPRINT is to be suppressed. This includes the suppression of any errors or warnings issued by the IEBUPDTE utility.

IMS Data Set Options

MACLIB=

Specifies the macro requirements of the generated system. This operand is used by system definition to determine the macro instructions to be moved into the IMS.MACLIB data set. This operand is ignored if the IMSCTRL macro SYSTEM operand specifies CTLBLKS or NUCLEUS. The default is UTILITY.

Attention: Do not change the MACLIB= specification after you have performed SMP/E JCLIN for your IMS system. The change is possible, but if the change is done incorrectly, it causes an unpredictable set of problems that are very hard to diagnose. A selection of MACLIB=ALL may be your best choice.

UTILITY

Specifies that only those macros necessary for PSB generation, DBD generation, and the dynamic allocation parameter list generation are copied to the IMS.MACLIB data set.

ALL

Specifies that all IMS macros, except those necessary for an IMS system definition, are copied to the IMS.MACLIB data set.

MACSYS=

Specifies the name for the MACLIB data set to be used in the stage 2 assemblies. The specified data set name cannot be longer than 44 alphanumeric characters.

NODE=

Specifies the node assigned to IMS data set names.

node1

Specifies the node to be assigned to all IMS data set names, as related to the use and generation of the data set names, by IMS system definition (see the list below). The specified node can contain from 1 to 26 characters, including imbedded periods. The first character of each level of qualification must be a letter, @, \$, or #. The remaining seven characters can be any of the preceding characters and the numerals zero through nine. The node name must conform to MVS data set naming conventions. The default node generated is IMS.

node2

Specifies the node to be assigned to the IMS data set names in the list below. This node overrides the *node1* assignment for these specific data sets. The character set restrictions are as defined for *node1*.

node3

Specifies the node to be assigned to the IMS data set names in the list below. This node overrides the *node1* assignments for these specific data sets. The character set restrictions are as defined for *node1*.

node1	node1	node2	node3
MSDBCP1	IMSL0G2	PROCLIB	GENLIB
MSDBCP2	RDS	RESLIB	GENLIBA
MSDSCP3	RDS2	MACLIB	GENLIBB
MSDBCP4	PGMLIB	MATRIX	LOAD
MSDBDUMP	PSBLIB	MATRIXA	DBSOURCE
MSDBINIT	DBDLIB	MATRIXB	SRCOURCE

QBLKS	ACBLIB	JOB	LGENIN
QBLKSL	ACBLIBA	OPTIONS	LGENOUT
SHMSG	ACBLIBB	MODBLKS	SVSOURCE
SHMSG	REFERAL	MODBLKSA	TMSOURCE
LGMSG	FORMAT	MODBLKSB	
LGMSG	FORMATA		
MODSTAT	FORMATB		
MODSTAT2	TFORMAT		
IMSMON	SYS0nnn		
IMSLOG			

OBJDSET=

Specifies the name of a cataloged partitioned data set into which assembler object modules are assembled during stage 2 of IMS system definition. The specified data set name must not be longer than 44 alphanumeric characters. If this parameter is not supplied, these modules are placed in IMS.OBJDSET.

PROCLIB=

Specifies whether system procedures are to be generated. If PROCLIB=NO, no system procedures are generated. If PROCLIB=NO, a new generation does not reflect the proper defaults because the newly generated procedures are not stored in PROCLIB. Therefore, care should be taken when specifying this option.

If PROCLIB=IMS and a CTLBLKS, NUCLEUS, or ON-LINE generation is specified in the IMSCTRL macro statement, only the IMS procedure is generated. If PROCLIB=IMS is specified and a BATCH or ALL generation is specified, the PROCLIB=IMS specification is ignored and the default (YES) is used.

The second parameter (n) is no longer supported by SYSGEN. It was used to define the OSAM buffer pool size in 1K increments for a DL/I batch region. If coded for compatibility, it is accepted and entered as the BUF= parameter on the DBBATCH, DLIBATCH, IMSCOBGO, and IMSPLIGO cataloged procedures.

If the second parameter is not coded, a default value of 7K bytes is used. BUF= is generated in the cataloged procedures and entered as the BUF= parameter in the DBBATCH, DLIBATCH, IMSCOBGO, IMSPLIGO catalog procedures.

For information on specifying database buffer pool size batch region for a DL/I batch region, see "Specifying IMS Buffer Pools" on page 378.

USERLIB=

Specifies the name of the library in which user-written routines, such as message edit routines, are to be included in the generated IMS nucleus. If this operand is omitted, the library containing the routines is assumed to be the same as that named on the RESLIB DD statement. The name of the data set, which must be cataloged, must not be longer than 44 characters.

MODGEN=

Specifies the name for the MODGEN data set to be used in the stage 2 assemblies. The specified data set name cannot be longer than 44 alphanumeric characters.

UMAC0=

Specifies the name of a data set to be put at the top of the SYSLIB concatenation of the stage 2 assembly job steps. The specified data set name cannot be longer than 44 alphanumeric characters.

IMSGEN Macro

UMAC1=,UMAC2=,and UMAC3=

Specifies the name of a data set to be concatenated to the SYSLIB DD statement of the stage 2 assembly job steps. The specified data set name cannot be longer than 44 alphanumeric characters.

Typically SYSMAC is used for either SYS1.MACLIB or SYS1.AMACLIB, and MODGEN is used for either SYS1.MODGEN or SYS1.AMODGEN. If the SYS1.AMACLIB and SYS1.AMODGEN data sets are specified, the UMAC1= keyword must specify SYS1.ATSOMAC. This insures that all macros that might be referenced are at the same level.

If no value is provided, the corresponding statement is dropped from the SYSLIB DD.

Job Control Language Statements

Use the first two parameters (JOBCTL and PRTY) or the last two (JCL and SCL).

JOBCTL=

The following parameters are valid:

nbr

Specifies the number of steps per JOB statement to be produced by system definition stage 1 for execution of stage 2. The maximum allowable value is 255; the default is 5. Regardless of the value specified, a JOB statement is produced for the beginning of the link-edit steps and for the nucleus link-edit step (if applicable).

jobclass

Specifies the job class to be generated on the stage 2 JOB statement. The default is A.

outputclass

Specifies the output class to be generated for the stage 2 JCL. The default is A.

job accounting and/or step accounting

Specifies job and/or step accounting data to be placed in the stage 2 JCL. The length of each set of (job and step) accounting data cannot exceed 50 bytes. If job accounting data is specified, a programmer name of IMS is provided.

When the accounting information fields contain special characters (except hyphens), special considerations apply to the manner in which this field is specified. See "Sample IMSGEN Macro Statement" on page 107.

PRTY=

Specifies the priority placed on the JOB statements for IMS system definition stage 2 jobs. The default is priority 0.

JCL=

The following parameters are valid:

jobname

Specifies a maximum of six alphanumeric characters to be used as the first portion of the generated job names. The last two characters of the job names are internally generated, sequentially incremented, hexadecimal numeric values representing the relative position of each job in the stage 2 stream. The default is IMSGEN.

The character set that can be used is described on the node keyword.

job accounting

Specifies job accounting data to be placed in the stage 2 JCL. The length of the accounting data cannot exceed 50 bytes.

When the accounting information fields contain special characters (except hyphens), special considerations apply to the manner in which this field is specified. See "Sample IMSGEN Macro Statement" on page 107.

programmername

Specifies the programmer name to be placed in the stage 2 JCL. The default is IMS.

outputclass

Specifies the output class to be generated for the stage 2 JCL. The default is A.

job miscellaneous 1 & 2

Specifies any additional parameters the user can desire to have placed in the stage 2 JOB statements. Length of this parameter cannot exceed 60 bytes. If you code this parameter, do not code the MSGCLASS parameter.

MFSDFMT=

Specifies whether (YES) or not (NO) the default message format screens are built in stage 2 of IMS SYSGEN. If MFSDFMT=NO, stage 2 does not have the default MFS screens generated in stage 2. The default is YES.

ONEJOB=

Subparameter1 specifies whether (YES) or not (NO) a single job is to be built for stage 2. The default is NO. If ONEJOB=NO, stage 2 has a multiple of job steps as specified in JOBCTL= subparameter 1.

Subparameter 2 specifies whether (YES) or not (NO) condition code checking for the link-edit steps is to be performed. If ONEJOB=(YES,YES) is specified, condition code checking is done in the link-edit steps. These steps are bypassed if any previous step did not complete with a condition code of 0. The default for subparameter 2 is YES if subparameter 1 is YES.

When ONEJOB= is specified for a single JOB statement, JOBCTL= and SCL= subparameter 1 are ignored.

If condition code checking is being specified for SCL= and ONEJOB= subparameter 2 is YES, a JCL error occurs due to the specification of duplicate COND= parameters.

SCL=

The following parameters are valid:

nbr

Specifies the number of steps per JOB statement to be produced by system definition stage 1 for execution of stage 2. The maximum allowable value is 255; the default is 5. Regardless of the value specified, a JOB statement is produced for the beginning of the link-edit steps and for the nucleus link-edit step (if applicable). This parameter is ignored in two cases:

- The first job that updates the IMS options data set or the IMS PROCLIB has one or two job steps for system definitions (one job step for each applicable update). IMS requires this for compatibility with the SMP/E GENERATE function.

IMSGEN Macro

- The last job is a multi-step job with two or four steps (for system definitions that create MFS libraries. This is caused by steps 2 and 4 (if present) being dependent on the respective preceding step.

step accounting

Specifies step accounting data to be placed in the stage 2 JCL. The length of the accounting data cannot exceed 50 bytes.

When the accounting information fields contain special characters (except hyphens), special considerations apply to the manner in which this field is specified. See "Sample IMSGEN Macro Statement" on page 107.

step miscellaneous

Specifies any additional parameters that you might want to place in the stage 2 EXEC statements. The length of this parameter cannot exceed 50 bytes. Do not code REGION= here, because internal IMS processing automatically generates this parameter. Do not code COND= if the MFSDfmt=NO keyword is not coded. Some of the default MFS job steps have COND= already coded.

UJCL1=,...,UJCL5=

Specifies the contents of a JCL statement that is placed between the JOB statement and the first EXEC statement in the stage 2 job stream. The specified statement cannot be longer than 60 characters. No other syntax checking of the parameter contents is done.

The intent of these keywords is to provide comment or JES control statements in the stage 2 job stream. These parameters can be used, for example, to add ROUTE statements, JOBLIB DDs, or JOBCAT DDs.

If no value is provided, the corresponding statement is dropped from the SYSLIB DD.

General Communication Options

All communication operands specified are ignored and a warning message is issued if the COMM macro is specified. See the section entitled "Relationships between the SECURITY, COMM, and IMSGEN Macros" on page 137.

MFSTEST=

Specifies whether (YES) or not (NO) the Message Format Service test facility, MFSTEST, is to be included in the generated system. YES is invalid for systems not containing MFS terminals. The default is NO.

Be aware that the use of MFSTEST can degrade IMS performance due to MFSTEST use of the communication line buffer pool.

SYSMMSG=

Specifies whether (TIMESTAMP) or not (NOTIMESTAMP) the system message timestamp facility is to be included for the generated system. The default is NOTIMESTAMP.

When TIMESTAMP is specified, the time that a message was generated is inserted between the message number and the message text for each message in the table.

Related Reading: Refer to the DFS messages introduction in *IMS/ESA Messages and Codes* for more information.

Security Options

All security options are ignored or overridden if the COMM or the SECURITY macro is specified.

PSWDSEC=

Specifies whether (YES or FORCE) or not (NO) password security is to be generated as the default for the /NRESTART command when issued to start the IMS system. The default is NO.

FORCE specifies that the IMS MTO cannot negate password security when issuing the /NRESTART command. If YES is specified, the MTO can override password security inclusion by specifically requesting no security on the /NRESTART command.

SECCNT=

Specifies the maximum number of terminal and password security violations to be accepted per physical terminal and transaction command violations per transaction prior to master terminal notification of such violation. The default is 0, which nullifies notification to the master terminal. The number specified can range from 0 through 3.

If SECCNT is not 0, the master terminal is notified for every violation.

TERMSEC=

Specifies whether (YES or FORCE) or not (NO) terminal security is to be generated as the default for the /NRESTART command when issued to start the IMS system. The default is NO.

FORCE specifies that the IMS MTO cannot negate terminal security when issuing the /NRESTART command. If YES is specified, the MTO can override terminal security inclusion by specifically requesting no security on the /NRESTART command.

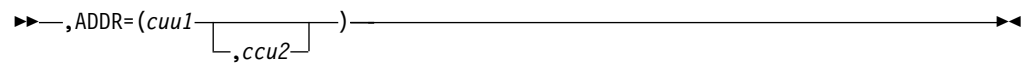
Sample IMSGEN Macro Statement

The following example of the IMSGEN macro statement indicates that:

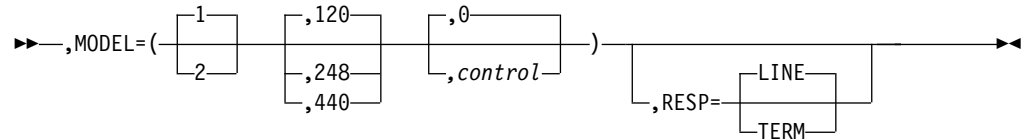
- High Level Assembler is used.
- An assembly listing is desired.
- Linkage editor print options are MAP and LIST.
- SIZE parameter specification is not required.
- Linkage editor region size is 130K.
- Composite control block, nucleus, map module, nucleus and security directory blocks module suffix is 9.
- All IMS macros (except those required for system definition) are to be copied to the appropriate MACLIB.
- The cataloged partitioned data set into which assembler load modules are to be placed is IMSXXX.OBJ.
- The library containing user routines is IMSXXX.USER.
- System procedures are requested.
- All IMS data set names are to be prefixed by the node IMSXXX (that is, IMSXXX.RESLIB).

```

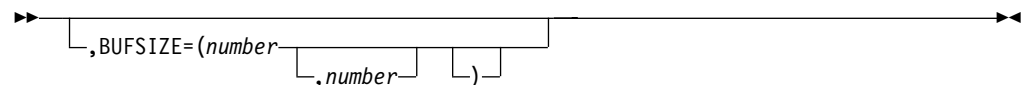
IMSGEN  ASMPRT=ON,LKPRT=(MAP,LIST),SUFFIX=9,      X
        MACLIB=ALL,OBJDSET=IMSXXX.OBJ,           X
        USERLIB=IMSXXX.USER,                     X
        PROCLIB=YES,NODE=IMSXXX
    
```

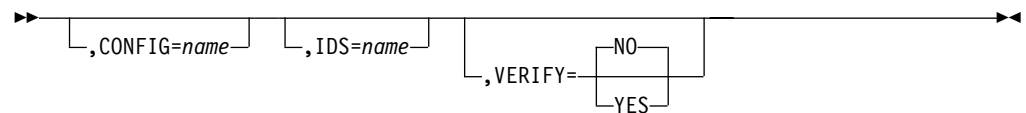
2740 Lines Only



3270 (Local), S/3, S/7, Disk Tape, and Spool Lines Only



3275 Switched Lines Only



Keyword Parameters

To find which parameters apply to your IMS configuration, refer to Table 12 on page 32.

ADDR=

Specifies the address of the communication line as defined in the Transmission Control Unit. The address value is three hexadecimal digits between 000 and 6FF for MVS/ESA SP V4. The address value is three or four hexadecimal digits between 0000 and FFFF for MVS SP V5 or later. All line addresses specified in the system definition must be unique values. This operand is used only to generate IMS DD statements for cataloged procedures. It is not allowed if the terminal is a 3270 local. This operand is not required for SPOOL line groups.

This operand is ignored for UNITYPE=PRINTER when a JES spool data set (SYSOUT=X) is to be used rather than a local printer. When XRF is used, a second optional device address can be specified with the ADDR keyword. This operand defines the second address for a BTAM-supported IMS master or secondary master terminal. And, when specified, only the master or secondary master terminals can be in the line group.

BACKUP=

Specifies (for XRF only) the control of the automatic restart after takeover. Use only when HSB=YES is specified on the IMSCTRL macro.

X is a numeric integer from 1 (lowest) to 7 (highest), inclusive, that specifies the priorities for reestablishing the session. If the entire keyword or the operand is omitted, the default is 4. Specifying BACKUP=NO suppresses the automatic restart of the devices after a takeover.

LINE Macro

BUFSIZE=

This operand is only valid for 3270 (LOCAL), DISK, TAPE, System/3, System/7, and SPOOL lines.

For 3270 local lines, this operand is used to specify the maximum size of an input message on this line. The operand is required for 3270 local lines to which 3277 terminals are attached and is not allowed for lines to which 3284 or 3286 printers are attached. The minimum specification is 300 bytes; the maximum specification is 32000 bytes.

For DISK, TAPE, or SPOOL lines, this operand is used to specify, in bytes, the maximum output buffer size to be used for the line. The operand is required. The minimum specification is 16; the maximum specification is 32760.

For System/3 and System/7 lines, this operand is used to specify the input and output buffer sizes to be used for this line. The first parameter is the input buffer size; the second parameter is the output buffer size. The operand is required. The minimum specification for either parameter is 50; maximum specification is 65000.

For System/3 and System/7 BSC lines, the input buffer size specified is doubled by IMS to allow handling of reverse interrupts.

For System/7 lines, the output buffer size specified is doubled by IMS to allow for conversion to pseudo-binary PTTC/EBCD unless the user specifies FEAT=PTTC/EBCD in the STATION macro. The input buffer size, however, is not automatically doubled by IMS. It is the responsibility of the user to ensure that the buffer size specified contains the data in the format specified by the STATION macro.

CONFIG=

Specifies the name of a CONFIG macro, previously defined in either this line group or in a previously defined line group, that describes the 3275 configuration to be used:

- When VERIFY=NO is specified and either no IDLIST macro is provided or one or more of the list entries does not name a configuration to use.
- When VERIFY=YES is specified and one or more of the list entries does not name a configuration to use.

If CONFIG= is not specified, the IMS default configuration is used unless IDS= is specified.

IDS=

Specifies the name of the IDLIST macro, previously defined either in this line group or a previously defined line group, that is to be used for this line. When IDS= is used, no subsequent IDLIST macro need be specified for this line. The CONFIG specification from the named IDLIST macro's associated LINE macro is used unless this line also specifies CONFIG=.

MODEL=

Specifies (for 2740 line groups only) the model. The first parameter specifies the model number of the terminals attached to this line. If MODEL=1 is specified, all other parameters of the operand must be omitted. MODEL=2 is invalid for switched 2740 line groups. The default is 1.

The second parameter specifies the maximum input buffer size for all 2740 Model 2 terminals on this line. The default value is 120.

The third parameter specifies the header control feature for 2740 Model 2 terminals on this line, and the number of characters reserved for header control. The default value is 0. The value for header control must not exceed 28, and it must be a multiple of 4.

RESP=

Specifies that 2740 station control and 2740 Model 2 terminals are to operate in either line (LINE) or terminal (TERM) response mode. All terminals that are eligible to enter Fast Path transactions must operate in terminal response mode (TERM).

VERIFY=

Specifies whether (YES) or not (NO) terminal hardware ID verification should be performed when IMS receives a dialed-in call from a switched 3275. The default is NO. If VERIFY=NO, IMS checks for the presence of a list of terminal IDs (see "IDLIST Macro" on page 78). If no list is found, IMS sets up the configuration specified in the LINE macro (CONFIG=); if CONFIG= is not specified, IMS sets up the default configuration (see "CONFIG Macro" on page 73). If IMS finds an ID list, it searches the list for the calling terminal. If no match is found, IMS sets up the configuration specified in the LINE macro or the IMS default. If a match is found, IMS sets up the configuration accordingly.

If VERIFY=YES is specified, IMS searches the list of terminal IDs (established by the IDLIST macro) to determine if the calling terminal is authorized to access IMS using this line. If IMS finds a match, it sets up the configuration associated with this terminal (see "CONFIG Macro" on page 73). If IMS does not find a match, it disconnects the terminal, sends the DFS2106 UNKNOWN TERMINAL TRIED TO CONNECT ON LINE X PTERM Y message to the master terminal, and creates a X'69' log record containing the terminal ID and the number of the accessed line.

After the configuration is established, IMS sends a DFS073 /IAM COMMAND REQUIRED message to the calling terminal. The /IAM command is required to establish a logical connection to IMS.

Example: An example of the LINE and LINEGRP macros used in combination to define a 2740 Model 2 with maximum input buffer of 248 bytes, line address of 024, and no header control or edit routine is:

```
LINEGRP DDNAME=D27402,UNITYPE=2740
LINE    ADDR=024,MODEL=(2,248)
```

Use of SYSIN Local Card Reader: When the DD statement for a line (specified in the LINEGRP macro statement as UNITYPE=READER) does not allocate a card reader device, the following conditions apply:

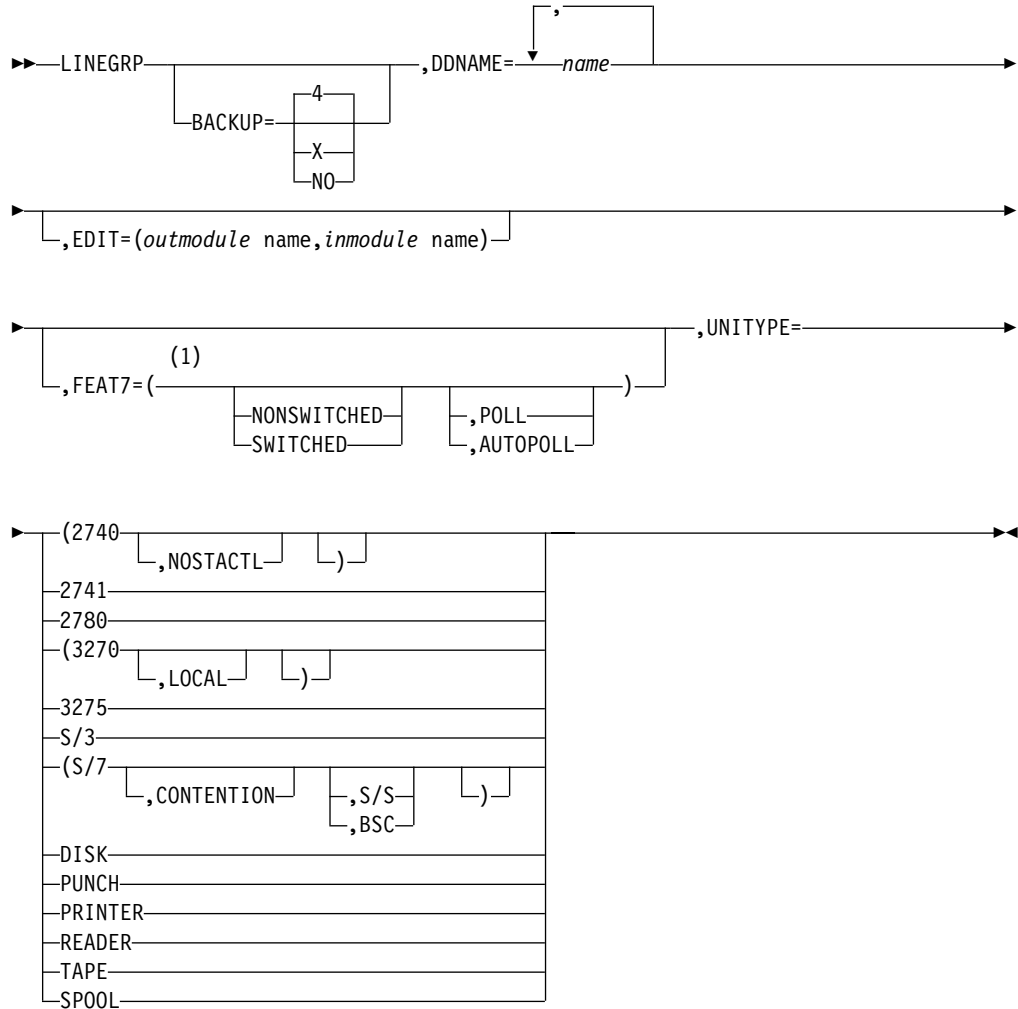
- If IMS is running as a batch program, the DD statement referring to the line must contain DCB=BLKSIZE=80.
- If IMS is running as a system task, the SYSIN test stream must be placed in a sequential data set with the DCB attributes RECFM=F and BLKSIZE=80. The SYSIN test stream must be pointed to by a DD statement for a reader line within the IMS procedure.

Although VTAM requests are priority-ordered by IMS, the active requests can be completed in any order because of internal VTAM conflicts and pacing.

LINEGRP Macro

The LINEGRP macro statement defines the beginning of a set of macro instructions that describe the user's telecommunications system. This macro statement begins a description of one or more lines of the same type over which the same type of terminal communicates.

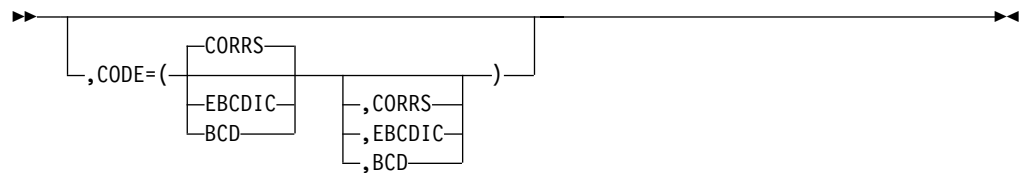
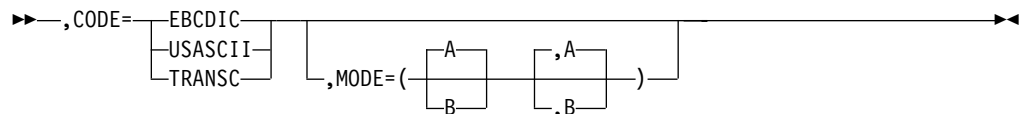
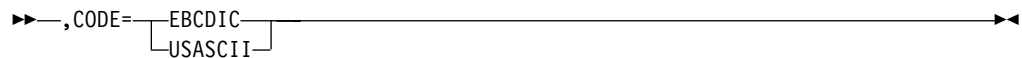
All Line Groups



Notes:

- 1 NONSWITCHED is the default except when a device that is not a valid nonswitched device is specified by UNITYPE=.

2741 Nonswitched Line Groups Only

**2741 Switched Line Groups Only****2780 Line Groups Only****3270 Remote and 3275 Switched Line Groups Only**

Keyword Parameters

To find which parameters apply to your IMS configuration, refer to Table 12 on page 32.

BACKUP=

Specifies (for XRF only) the control of automatic restart after takeover. Use only when HSB=YES is specified on the IMSCTRL macro.

X is a numeric integer from 1 to 7, inclusive, which specifies the priority assigned to reestablishing the session. When either the keyword or the operand is omitted, the default is 4. Specifying BACKUP=NO suppresses the automatic restart of the devices after a takeover.

CODE=

Specifies translate options for 2780 line groups, 2741 line groups, and 3270 remote line groups only.

For 2780 line groups, specifies translate table EBCDIC/EBCDIC (EBCDIC), RASA/SASA (USASCII), or RC80/SD80 (TRANSC). The operand is required.

For nonswitched 2741 line groups, specifies translate table SD40/RC40 (EBCDIC), SB40/RB40 (BCD), or SD41/RC41 (CORRS). The default is CORRS. The operand is required.

LINEGRP Macro

For switched 2741 line groups, specifies one or a combination of translate tables. Actual translate table determination is performed when the terminal calls IMS. The combination of EBCDIC and BCD is invalid. The default is CORRS (RC41/SD41).

DDNAME=

Specifies a one- to eight-character name that associates the generated DCB for this line group, within the IMS nucleus control block, with the generated JCL DD statements identifying the lines within the line group. If the line group being defined is a SPOOL line group, a list of up to 20 names can be specified: 255 SPOOL data sets, but no more than 20 names per line group can be specified. Each SPOOL name specification is associated, in the order in which it was specified, with the sequentially generated SPOOL data set names produced for the SPOOL print procedures. These procedures are generated and named sequentially for each SPOOL line group. Refer to "LINEGRP Macro" on page 112, for an example of SPOOL data definition name association.

The names must begin with an alphabetic character. The operand is required. The following names cannot be used as LINEGRP ddnames:

DFSRESLB	IMSLOG	IMSTFMT	MSDBINIT
DUMP	IMSLOGR	IMSUDUMP	PRINTDD
IEFRDER	IMSLOGR2	LGMSG	PROCLIB
IEFRDER2	IMSLOG2	MATRIX	QBLKS
IMSACB	IMSMON	MSDBCP1	SHMSG
IMSDBL	IMSRDS	MSDBCP2	
IMSDILIB	IMSSPA	MSDBDUMP	

Recommendation: Allocate at least two data sets.

EDIT=

Specifies a one- to eight-character, user-supplied physical terminal output edit routine and a one- to eight-character, user-supplied physical terminal input edit routine for the terminal type in this line group.

Related Reading: For more information, refer to *IMS/ESA Customization Guide*.

This routine cannot be the same as the one that is used on a TRANSACT EDIT= parameter.

FEAT=

Determines whether this group of telecommunication lines is leased (NONSWITCHED) or not leased (SWITCHED). The default is NONSWITCHED, except for devices that are not valid nonswitched devices. SWITCHED is forced for 3275³

IMS supports the IBM Communicating Magnetic Card/ELECTRIC. Typewriter (CMC/ST) as a switched 2741. Therefore, when a CMC/ST is used, it must be defined to IMS as a switched 2741. CMC/ST features that are not compatible with 2741 features, and that result in different operations, are not supported.

Polling features are ignored for READERS, PRINTERS, PUNCHes, TAPes, DISKS, SPOOLs, System/3s, and 3270s. The AUTOPOLL polling feature is invalid for 3270s (LOCAL), 2741s, and 2740s (NOSTACTL). No polling list is generated for 3270s (LOCAL), 2740s (NOSTACTL), 2741s, READERS,

3. A 3275 on a multidropped line is defined on the LINEGRP macro statement as UNITYPE=3270 and on the TERMINAL macro statement as UNIT=3275.

PRINTERS, PUNCHes, TAPES, DISKs, or SPOOLs. A BTAM AUTOWLST is generated for 3270s and multipoint 2780s. No polling list is generated for point-to-point 2780s unless specifically requested by specifying AUTOPOLL.

Direct SYSOUT specifications (PRINTER, PUNCH, TAPE, or DISK) do not require that specific device type be assigned at IMS execution time. All specifications except PRINTER affect only default BUFSIZE or generated JCL. Specifying PRINTER results in execution time being translated to the 48-character set, lowercase is translated to uppercase, and all other codes translated to periods(.). If a specification other than PRINTER is used, no translation occurs. If a line not generated as a PRINTER is allocated to a printer having the universal character set feature (UCS), and fold mode operation is used, unprintable characters print as extraneous alphanumerics.

MODE=

The first parameter specifies that communication is to be through the 270x Data Adapter Unit's Dual Communication Interface.

The second parameter specifies the use of the transmission code designated by Code (A or B) for the 270x Data Adapter Unit Dual Code Feature. This feature must be present on the 270x. A is the default.

UNITYPE=

Specifies the terminal device type contained in this line group. If UNITYPE=(3270,LOCAL) is specified, only one LINE statement can be in the line group.

CONTENTION (as opposed to a polled line) is a valid specification for System/7 only. Either S/S (start/stop, the default) or BSC (binary synchronous) can be specified.

When SWITCHED is specified for FEAT=, only 2740 with station control, 2741, and 3275 are valid for the UNITYPE= parameter. ³

When FEAT=NONSWITCHED, valid UNITYPE parameters are 2740 with or without station control, 2741, 2780, 3270 local and remote, S/3, S/7, DISK, PUNCH, PRINTER, READER, TAPE, and SPOOL.

If any of the following UNITYPE parameters are specified, BTAM is not included in the IMS system: DISK, PUNCH, PRINTER, READER, TAPE, or SPOOL.

When a JES SPOOL DATASET (SYSOUT) is used, UNITYPE = PRINTER must be specified.

Example: Following is an example of the LINEGRP macro for a 2780 line group with no output edit routine and EBCDIC translation:

```
LINEGRP DDNAME=DD2780,UNITYPE=2780,CODE=EBCDIC
```

An example of the LINEGRP macro and the resulting ddname/dsname and print procedure associations for a SPOOL line group is:

```
LINEGRP DDNAME=(NAME1,NAME2),UNITYPE=SPOOL
```

```
LINEGRP DDNAME=(NAMEA,NAMEB),UNITYPE=SPOOL
```

The SPOOL print procedure for line group 1 is named IMSWT000. The data set names produced for this print procedure are IMS.SYSO1 and IMS.SYSO2.

LINEGRP Macro

The SPOOL print procedure for line group 2 is named IMSWT001. The data set names produced for this print procedure are IMS.SYSO3 and IMS.SYSO4.

The corresponding JCL statements generated for the IMS online execution procedure are as follows:

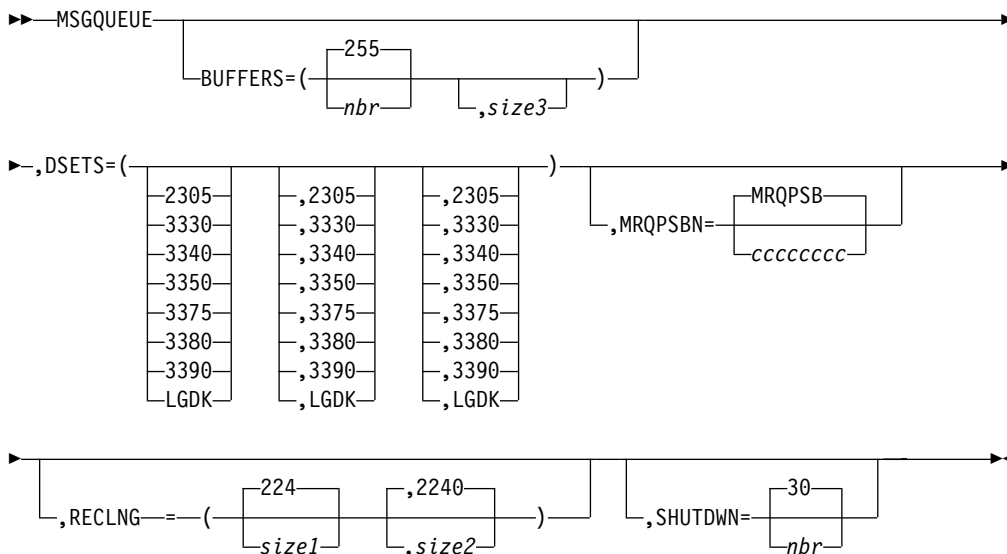
```
//NAME1 DD DISP=SHR,DSNAME=IMS.SYSO1
//NAME2 DD DISP=SHR,DSNAME=IMS.SYSO2
//NAMEA DD DISP=SHR,DSNAME=IMS.SYSO3
//NAMEB DD DISP=SHR,DSNAME=IMS.SYSO4
```

If UNITYPE=SPOOL on the LINEGRP macro statement, ddnames used in the preceding JCL must agree with the actual ddnames specified on this statement.

MSGQUEUE Macro

You must include the MSGQUEUE macro statement when the type of generation specified in the IMSCTRL macro statement is ALL, ON-LINE, CTLBLKS, or NUCLEUS. This statement defines the characteristics of the three message queue data sets (IMS.QBLKS, IMS.SHMSG, and IMS.LGMSG).

The information you specify in this macro is also used in a shared queues environment.



Keyword Parameters

To find which parameters apply to your IMS configuration, refer to Table 12 on page 32.

BUFFERS=

Specifies the number of buffers to allocate for message queue management, and the block size to be assigned for all three message queue data sets. If *nbr* is not specified, a default of 255 is used. The minimum *nbr* specification is 3; the maximum is 9999. If *size3* is not specified, it is calculated by the following formula:

$$\text{SIZE3} = ((\text{SIZE1} + \text{SIZE2} - 1) / \text{SIZE1}) \times ((\text{SIZE1} + 55) / 56) \times 56$$

For the division operations in this formula, only the integer portion of the result is used; the fractional portion of the result is truncated.

This calculation can leave null space at the end of the buffer block for short and long message records.

size1 and *size2* should be an even multiple of the QBLKS LRECL of 56.

If the RECLNG defaults of *size1* (224) and *size2* (2240) are used, then *size3* of BUFFERS equals 2240. The maximum specification for *size3* is 30632, or the track length of the device on which the data set resides, whichever is smaller. All sizes specified are rounded up to multiples of 4.

RECLNG can be changed prior to a cold start of IMS, but if a smaller value is specified prior to a BLDQ restart, the restart might abend.

You should not allocate less space for a restart of IMS than was allocated for the prior execution. If a smaller logical record length has been allocated, or if less space for the Message Queue data sets has been allocated for restart with BUILDQ, the restart process may terminate abnormally.

DSETS=

Specifies the device types on which the three message queue data sets (IMS.QBLKS, IMS.SHMSG, and IMS.LGMSG, respectively) are to reside. If all three data sets are to reside on the same device type, you can specify just the first parameter.

You can specify the following device types: 2305, 3330, 3340, 3350, 3375, 3380, 3390, or LGDK.

When using a 3350, specify the drive format used—3330 or 3350.

LGDK is a generic definition for disk drives that have a track size equal to or greater than 32767 bytes. The 3375, 3380, 3390, and future devices can also be defined as LGDK.

MRQPSBN=

Specifies the one- to eight-character alphanumeric name of the Message Requeuer Program Specification Block. Specification of this parameter permits IMS to use the support provided by the IMS/ESA Message Requeuer licensed program 5655-136. The default name is MRQPSB. Regardless of whether you use the default name, you must include a PSB for the message region in the stage 1 gen. If you do not use the default name, you must specify a name for the Message Requeuer Specification Block with the MRQPSBN keyword. If an error is made in specifying the Message Requeuer PSB name, a warning message is issued and the default name is used.

RECLNG=

Specifies the logical record lengths for the short and long message queue data sets, respectively.

The minimum value for *size1* is 112. The value specified for *size1* must be large enough to accommodate the ISC/LU6.1 (type 84) prefix and the Conversation Extension (type 8D) prefix. If Shared Queues is used, it must also include an additional X'28' bytes.

MSGQUEUE Macro

The minimum value for *size2* is 672. *size2* must be equal to or greater than *size1*. If either *size1* or *size2* is not specified, defaults of 224 and 2240, respectively, are assigned. The value of *size2* cannot exceed the track length of the device on which the data set resides.

The *size2* value must be divisible by 4; otherwise the MSGQUEUE macro rounds it up until it is. *size1* and *size2* values should be even multiples of the QBLKS LRECL of 56.

If a message does not fit in a large queue record (*size2*), IMS either spans it to other queue records, or rejects the message with an error.

Some messages (such as MFS) from an input terminal can span message queue records. If messages span message queue records, the size of the input segment has no limit. When messages do not span message queue records, the maximum allowable size for input segments is *size2* minus the amount for various prefix items included with the data portion of the message queue record. The maximum size allowed for output segments is *size2* minus the amount for various prefix items included with the data portion of the LRECL (message queue record). Prefix items and sizes are displayed in the following table:

Table 17. Message Prefix Size

Prefix Section	IMS V4 Size (in bytes)	IMS V5 Size (in bytes)	IMS V6 Size (in bytes)	Comments
Basic ¹	20	20	64	All messages
System Segment (81)	56	56	56	All messages
	64	64	64	All messages with MFS
MSC(82) ²	64	64	N/A	All input and all remote output messages, if MSC is sysgened
Security Prefix (83) ³	8	8	8	All messages if RACF or ETO is sysgened
ISC/LU6.1 Prefix (84) ⁴	22	22	22	All ISC/LU6.1 messages
APPC (LU6.2) (85) ⁵	128-512	N/A	N/A	All APPC messages
Extended Prefix (86) ⁶	N/A	16	16	All messages
APPC (LU6.2) (87) ⁵	N/A	128-512	128-512	All APPC messages
OTMA (87) ⁵	N/A	128-4096	128-4096	All OTMA messages
Security Prefix (88) ³	N/A	20	20	All messages if RACF or ETO is sysgened
Workload Manager (89)	N/A	16	16	All messages
System Extension (8A)	N/A	24	24	All messages
MSC Extension (8B)	N/A	40	92	All messages if MSC is sysgened
TMR (8C) ²	N/A	N/A	70	All messages
			102	
Conversation Ext. (8D) ⁶	N/A	N/A	40	All conversational messages

Table 17. Message Prefix Size (continued)

Prefix Section	IMS V4 Size (in bytes)	IMS V5 Size (in bytes)	IMS V6 Size (in bytes)	Comments
User Data Segments ⁷	V	V	V	All messages

Notes:

1. All messages begin with a basic prefix segment that has no prefix code. If the message is a first or only queue buffer message, it has a basic prefix and some or all of the prefixes type 81 thru 8D. User data segments can follow the IMS prefixes. For IMS V5 and V6, a message that is a spanned message queue buffer, but not the first queue buffer has only the basic prefix and user data segments. For IMS V4, if RACF is generated, the security prefix plus a short (4 bytes) segment extension is also included. Prefix segments 81 through 8D begin with a 3-byte LLC field where LL is the 2-byte length of the prefix segment (including the LL field) and C is the 1-byte prefix code.
2. The MSC prefix (type 82) is used in IMS V4 and V5 if MSC is sysgened. In IMS V6, it is converted to the transaction manager routing (TMR) prefix (type 8C) and is present in all messages
3. The security prefix (type 83) moved to the extended prefix area in IMS V5, as a type 88 prefix segment. In IMS V4, the security prefix is included in all messages. In IMS V5, the security prefix is included in only the first message buffer if RACF or ETO is used.
4. Because they are not sysgen values, the ISC/LU6.1 (type 84) prefix and the conversation extension (type 8D) prefix are always included in the maximum length of the IMS prefix calculated at IMS initialization. If Shared Queues is used, an additional 40 bytes is added.
5. The APPC prefix (type 85) moved to the extended prefix (type 87) in IMS V5. In IMS V5 and V6, both APPC and OTMA prefix segments use the type 87 prefix code. This prefix segment is present in all messages that originated from or are sent to an APPC or OTMA device.
6. All segments following the extended prefix header (type 86) are extended prefix segments. Extended prefix segments exist in IMS V5 and all later releases. The IMS release level determines the size of the extended prefix segments created for a message. When MSC or Shared Queues are used, a message can be processed by an IMS that is of a different release level than the originating IMS. If the extended prefix segments lengths are greater for the target IMS, IMS can create or increase the size of the extended prefix segments. Extended prefix segments are never shortened. Extended prefix segments are not sent to IMS V4.

Recommendation: In an environment where MSC and Shared Queues are used and IMSs are of different release levels, use the highest calculated values for *size1* and *size2* for each IMS.

7. User data segments follow the IMS prefix segments.

For messages that span message queue records, the first item (basic) is included in each record of the message. Also, if RACF is generated, the RACF item plus a small (4-byte) system segment is included. All other prefix items, if applicable, appear only in the first record of the message.

The values for *size1* and *size2* should be tuned for your system.

MSGQUEUE Macro

Related Reading: For information on how to tune the values of *size1* and *size2* after IMS is operational, refer to “Message Queue Data Set Allocation” in *IMS/ESA Installation Volume 1: Installation and Verification*, and to “Initial Optimizing of IMS Buffer Pools” in *IMS/ESA Administration Guide: System*.

The result of this tuning should yield a value for *size1* such that the I/O activity is equally split between the IMS.SHMSG data set, containing input and output message segments up to the length of *size1*, and the IMS.LGMSG data set, containing all input and output message segments larger than *size1*.

RECLNG can be changed prior to a cold start of IMS, but if a smaller value is specified prior to a BLDQ restart, the restart may abnormally terminate.

If emergency restart procedures using BUILDQ are to be used, reallocation of logical record and data set spaces must be done carefully. The BUILDQ procedure always restores the message queue entries to the relative position in the respective queue data sets they had at the time they were saved. If the logical record or data set size has been decreased, it may be impossible to perform the restart.

Related Reading: Refer to the information about restarting IMS in the *IMS/ESA Operations Guide* for additional details.

SHUTDOWN=

Specifies the number of records to be reserved in each data set to allow the system to automatically shut down if the data set becomes filled with unprocessed messages. The maximum valid specification is 32767. The default is 30. In an XRF system, this parameter also reserves the number of records in each of the local message queue data sets (IMS.LGMSG, IMS.SHMSG, or IMS.QBLKSL) for shutdown.

The number of records specified should provide enough space in each data set to allow for an orderly shutdown, which depends on message throughput and the number of regions that are scheduled. In general, the following calculation applies:

Maximum number of output messages between synchronization points per application \times number of regions scheduled + 1 input message per logical terminal active at the same time.

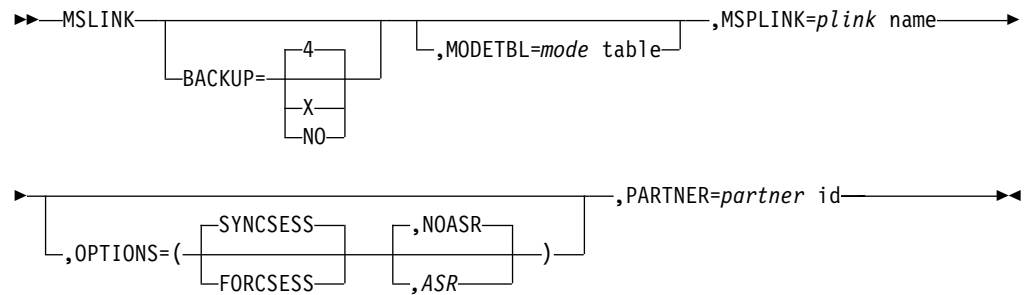
Example: An example of the MSGQUEUE macro statement with all data sets on a 3380 is:

```
MSGQUEUE DSETS=(3380),BUFFERS=(40,2240),SHUTDOWN=200
```

This example indicates that the default sizes for the short and long message queue data sets are used. 40 buffers are allocated; the size of each buffer block is 2240, allowing 1 long message record, 10 short message records, or 40 queue block records to fit into a block. 200 records in each data set are reserved to allow the system to automatically shut down.

MSLINK Macro

The MSLINK macro statement defines a logical link to another system. It can be followed by one or more macro statements that define the link name blocks (MSNAME) if any are to be defined for this logical link.



Keyword Parameters

To find which parameters apply to your IMS configuration, refer to Table 12 on page 32.

BACKUP=

Allows (for XRF only) the users to override switching options specified on the MSPLINK macro. It controls the automatic BTAM or VTAM Multiple Systems Coupling (MSC) link restart after takeover. Use only if HSB=YES is specified on the IMSCTRL macro.

X is a numeric integer from 1 to 7, inclusive. It sets priorities for reestablishing the session. When either the keyword or the parameter is omitted, the default is 4. NO suppresses the session recovery of the MSC physical link at takeover.

Although requests to VTAM are prioritized by IMS, the active requests can be completed in any order because of internal VTAM conflicts and pacing.

MODETBL=

Specifies the name of the VTAM logon mode table entry (logon mode name) containing the SNA bind parameters to be used when a session is established for this terminal. The maximum is 255 unique names for each IMS system.

This function allows a system definition specification for referencing an entry other than the default entry in the user's VTAM logon mode table.

With this function, if MODETBL= is not specified at system definition, no functional or operational change affects the user.

If MODETBL= is specified at system definition, the specified entry name is used.

The /DISPLAY command displays the current MODETBL name. The MODETBL name can be overridden by:

- VARY ACT, LOGON= command by the network terminal operator
- /RST or /CHANGE command by the master terminal operator

If the VTAM-node=name being defined is in another domain (that is, a cross-domain resource), the MODETBL parameter need not be specified.

Related Reading: See *Advanced Communications Function for VTAM (ACF/VTAM) Macro Language Guide* for additional information.

MSLINK Macro

MSPLINK=

This operand identifies the physical link to which this logical link is assigned. If this operand is not specified, no assignment to a physical link is made and the /MSASSIGN command must be used to assign the logical link to a physical link before communication can be established between the two systems.

When specified, the MSPLINK macro must be defined before the MSLINK macro.

OPTIONS=

This parameter allows you to specify whether session initiation should be completed. SYNCSESS indicates that session initiation is to be completed only if session resynchronization is successful. FORCSESS forces the session to be completed regardless of whether session resynchronization is successful. Successful session resynchronization occurs when the message sequence numbers of the two logical units in session agree, or when the sequence number of the sender is no more than 1 higher than the sequence number of the receiver. The default is SYNCSESS. The specifications made on this OPTIONS parameter must be consistent with the MSLINK OPTIONS specification in the partner systems. During IMS execution, this option can be overridden by the /CHANGE command.

The OPTIONS parameter is valid for MSC VTAM links only.

When using the Session Outage Notification facility, the second parameter allows you to specify automatic session restart (ASR) processing for a defined node. The default is NOASR.

Specifying ASR or NOASR on the MSLINK macro overrides ASR specifications on the TYPE and MSPLINK macros.

Use the /DISPLAY command to display the current ASR option, and the /CHANGE command to change it.

ASR works only if both sides of the link are using the ASR option.

PARTNER=

The partner-id is a two-character alphabetic identification. It ensures that the two related logical links in two systems are always logically and physically connected. Both systems must have MSLINK macro statements with the same partner-id. A logical link can be assigned to a different physical link; the two systems still communicate via the logical link, which remains as defined.

MSNAME Macro

The MSNAME macro statement provides a name for the remote and local system identifications that it represents. When used, it can be followed by one or more NAME macros defining remote logical terminals.

▶▶—MSNAME—SYSID=(*remote system id,local system id*)————▶▶

Label Field

linkname specifies a one-to eight- character alphanumeric name used externally in commands. Linknames, block names, transaction codes, and logical terminal names, collectively, cannot contain duplicates. Specifying the *linkname* is required.

Keyword Parameters

To find which parameters apply to your IMS configuration refer to Table 12 on page 32.

SYSID=

remote system identification identifies the remote system that is represented by this name.

The *local system identification* specifies the system identification that is to be used for routing messages back to this system.

remote system identification specification must not have been previously defined as:

- A remote system identification within a previous MSNAME macro statement.
- A local system identification within an APPLCTN, TRANSACT, or MSNAME macro statement.

local system identification specification must not have been previously defined as a *remote system identification* within a previous MSNAME macro statement.

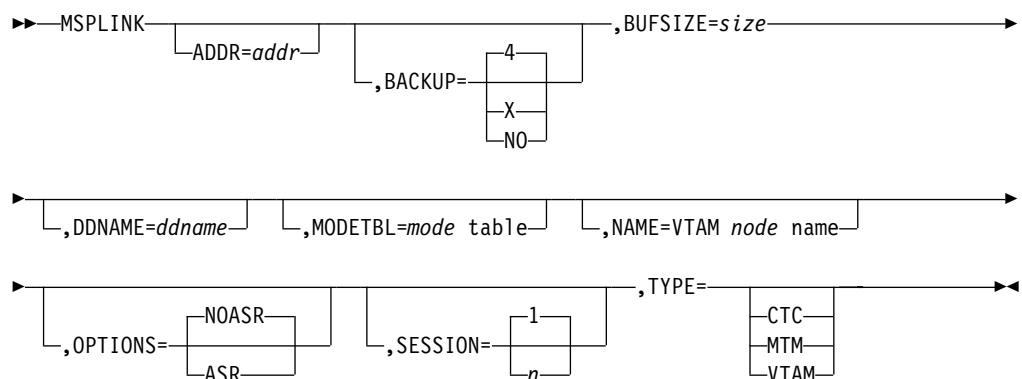
Values from 1 through 2036 are valid. A one-to-one relationship exists between remote SYSIDs and MSNAMEs. APPLCTN and TRANSACT macro statements must refer to remote SYSIDs specified in MSNAME statements in the same system definition when defining remote applications.

MSPLINK Macro

The MSPLINK macro statement defines a physical link. It can define three different types of connections between two systems: channel-to-channel connection (CTC), main-storage to main-storage connection (MTM), and ACF/VTAM.

Logical links are assigned to physical links during system definition (MSPLINK operand in MSLINK macro) or dynamically with /MSASSIGN commands.

Related Reading: Refer to *IMS/ESA Operator's Reference* for detailed information on the /MSASSIGN command.



MSPLINK Macro

Label Field

The MSPLINK name specifies a one- to eight-character alphanumeric name for this physical link. You must specify this name. This name is used (in the MSPLINK=keyword operand of the MSLINK macro) to assign a logical link to this physical link.

Keyword Parameters

To find which parameters apply to your IMS configuration, refer to Table 12 on page 32.

ADDR=

Specifies the address of the communication line or the address of the channel-to-channel adapter. The address is three hexadecimal digits for MVS/ESA SP V4 or earlier. The address value can be three or four hexadecimal digits for MVS/ESA SP V5 or later. This operand is required for CTC link types and invalid for MTM and VTAM link types.

All devices attached to the same channel as the CTC adapter must be accessible through an alternate channel. In addition, system resources (for example, paging devices) should not be attached to the same channel as the CTC adapter.

BACKUP=

Specifies (for XRF only) VTAM MSC link restart after takeover. Use only when HSB=YES is specified on the IMSCTRL macro.

X is a numeric integer from 1 to 7, inclusive, that specifies the priority for reestablishing the session. The default is 4 when either the keyword or the parameter is omitted. Specifying BACKUP=NO suppresses the automatic restart of the MSC physical link.

Although IMS prioritizes requests to VTAM, the active requests can be completed in any order because of internal VTAM conflicts and pacing.

BUFSIZE=

Specifies the input and output buffer sizes for each logical link defined for this physical link. For example, if three logical links are defined under a physical link specifying BUFSIZE=*nn*, six buffers each equal to *nn* (three input and three output) are allocated. The systems on each end of the physical link must specify equal buffer size. Consistency is checked during link startup. Minimum size is 160. For performance considerations, the specified buffer size should be 160 plus the larger of the maximum SPA size or maximum segment size.

MSC VTAM links require a minimum buffer size of 208 bytes and allow a maximum size of 30720 bytes.

LU 6.2 application transactions transmitted over MSC links need larger link buffers to accommodate the extra 500 bytes for the LU 6.2 message prefix and the IMS extended prefix. The buffer size must be a decimal value between 208 and 30720 that can be expressed by the algorithm X times 2 to the power of Y. X must be a value from 8 through 15, and Y must be a value from 3 through 11. This is a VTAM restriction. Thus, for example, the value 144 (representing 9×2^4) and the value 28672 (representing 14×2^{11}) are acceptable values.

Related Reading: Details of this VTAM restriction are found in *ACF/VTAM Macro Language Reference*.

This restricts the valid MSC VTAM buffer sizes to one of the following decimal values:

112	448	1792	7168	28672
120	480	1920	7680	30720
128	512	2048	8192	
144	576	2304	9216	
160	640	2560	10240	
176	704	2816	11264	
192	768	3072	12288	
208	832	3328	13312	
224	896	3584	14336	
240	960	3840	15360	
256	1024	4096	16384	
288	1152	4608	18432	
320	1280	5120	20480	
352	1408	5632	22528	
384	1536	6144	24576	
416	1664	6656	26624	

The value specified plus the size of any routing information later appended by VTAM must be less than the maximum message size specified for VTAM and its related NCPs.

DDNAME=

This is the ddname of the JCL statement that describes this physical connection. This operand is required CTC link types and is invalid for MTM and VTAM link types.

Each name must begin with an alphanumeric character. The following names cannot be used as ddnames:

DFSRESLB	IMSLOG	IMSTFMT	MSDBINIT
DUMP	IMSLOGR	IMSUDUMP	PRINTDD
IEFRDER	IMSLOGR2	LGMSG	PROCLIB
IEFRDER2	IMSLOG2	MATRIX	QBLKS
IMSACB	IMSMON	MSDBCP1	SHMSG
IMSDBL	IMSRDS	MSDBCP2	
IMSDILIB	IMSSPA	MSDBDUMP	

MODETBL=

Specifies the name of the VTAM logon mode table entry (logon mode name) containing the SNA bind parameters to be used when a session is established for this terminal.

MODETBL= allows a system definition specification for referencing an entry other than the default entry in the user's VTAM logon mode table. Normally, the terminal operator specified this mode table entry name when logging on to a terminal; this was not possible, however, from the IMS master terminal because IMS initiated the session.

If MODETBL= is not specified at system definition, no functional or operational change affects the user.

If MODETBL= is specified at system definition, the specified entry name is used.

The /DISPLAY command displays the current MODETBL name. The MODETBL name can be overridden by:

- The LOGON APPLID entry by the remote terminal operator
- VARY ACT, LOGON= command by the network terminal operator

MSPLINK Macro

- /OPNDST, /RST, or /CHANGE command by the master terminal operator

The MODETBL= parameter is required for the IMS master terminal if the VTAM default mode table has not been configured specifically for the device to be used as the IMS master terminal.

Specifying MODETBL on the MSPLINK macro circumvents the need to specify the mode table entry when logging on to terminals that always require specification of the same mode table entry name.

If the terminal being defined is in another domain (that is, a cross-domain resource), the MODETBL parameter need not be specified.

Related Reading: See *Advanced Communications Function for VTAM (ACF/VTAM) Macro Language Guide* for additional information.

NAME=

Specifies the VTAM node name of the remote system at the other end of the link. NAME= must be the same as the label on the VTAM APPL statement for the remote system (that is, the minor node name). For an MSC VTAM link communicating with an XRF complex, the node name must be the VTAM USERVAR associated with the partner IMS/XRF complex. If the ACBNAME parameter of the VTAM APPL statement is not specified, NAME= is the same as the application identification (APPLID=) specified on the IMS COMM macro statement in the remote system. This operand is required for MSC VTAM links only and is invalid for other link types.

OPTIONS=

The Session Outage Notification facility only, specifies automatic session restart on all logical links associated with physical links. The default is NOASR. ASR or NOASR in the MSLINK macro overrides ASR definitions on the MSLPINK macro.

Use the /DISPLAY command to display the current ASR option, and the /CHANGE command to change it.

ASR works only if both sides of the link are using the ASR option.

SESSION=

Specifies the number (from 1-676) of parallel sessions that can be active for the physical link defined by the MSPLINK macro statement. It is valid for MSC VTAM links only. The default is 1.

A large SESSION value enables the user to dynamically assign more logical links to the physical link than were originally assigned during system definition. The SESSION value can be increased at system startup by using JCL. For instructions, see the keyword NLXB described in "Chapter 4. Procedures" on page 189.

Because a given number of logical parallel sessions uses the same amount of storage for control blocks and buffers as would the same number of physical links, allocating many parallel sessions can use an excessive amount of common storage area. You should predetermine how much common storage area you want to use.

TYPE=

Defines the type of physical link being described. Valid parameters are CTC, MTM, and VTAM.

The MTM type link allows more than one IMS system to run in the same MVS/ESA system without a hardware link. It is intended for backup or testing.

NAME Macro

The NAME macro statement defines a logical terminal name (LTERM) associated with a physical terminal. Preparation of the NAME macro can be required for each of the macro sets that you prepare to define your IMS system.

The logical terminal can be specified as the IMS master terminal if the associated physical terminal is a 3740 or 3270 display terminal, SLU 1, or SLU 2. The terminal must be nonswitched. If the terminal is a SLU 1, both the input (ICOMPT) and the output (COMPT) component designations must refer to the first physical component. The first physical component of a SLU 1 terminal must be a console if it is to be defined as the IMS master terminal.

If the IMS master terminal is specified as a 2740 or SLU 1, the designation of a secondary master terminal is optional and can be specified if the logical terminal's associated physical terminal 2740 (but not an NTO device), or SLU 1. The terminal must be nonswitched. If the terminal is a SLU 1, both the input (ICOMPT) and the output (COMPT) component designations must refer to the first physical component. The first physical component of a SLU 1 can be either a console or a printer if it is to be defined as the IMS secondary master terminal.

If the IMS master terminal is specified as a 3270 display terminal, a secondary master terminal is required, and must be specified for a 328x, associated physical terminal.

If the IMS master terminal is specified as SLU 2, you must specify a secondary master terminal in a preceding NAME macro for a 2740, 328x, SLUTYPE1, or SPOOL associated physical terminal.

If other physical terminals are defined on the same communication line as the primary or secondary master terminal, then special restrictions will apply to the use of the /CHECKPOINT and /IDLE commands in the IMS system.

Related Reading: Details of these restrictions are contained in the corresponding command descriptions in *IMS/ESA Operator's Reference*.

The Message Format Service (MFS) special master terminal formatting is only available for 3270 display terminals specified with a screen size of 24 x 80.

For output purposes, the COMPT operand associates the specified logical terminal with a specific component in a 2780, 3275, 3601, SLU 1, or SLU P terminals, or LU 6.1 nodes. The COMPT operand is valid only for the above terminal types.

For SLU 1 terminals, the NAME macro statement also defines the data set at the terminal to receive output for corresponding components defined in the TERMINAL macro statement as USERDS1.

For LU 6.1 nodes, the position of the NAME macro within the system definition macro set determines whether LTERM allocation is fixed or dynamic.

If the allocation of LU 6.1 logical terminals is to be predefined (fixed), the NAME macro statement must follow the TYPE and TERMINAL macro statements, and the SESSION parameter must be equal to 1 (by specification or default).

NAME Macro

If the allocation of LU 6.1 logical terminals is to be dynamic, the NAME macro statement must follow the SUBPOOL macro statement, and the SESSION parameter must specify a valid value in the range 1 through 255. The default is 1.

If two or more sessions are defined on the SESSION parameter, dynamic allocation must be used.

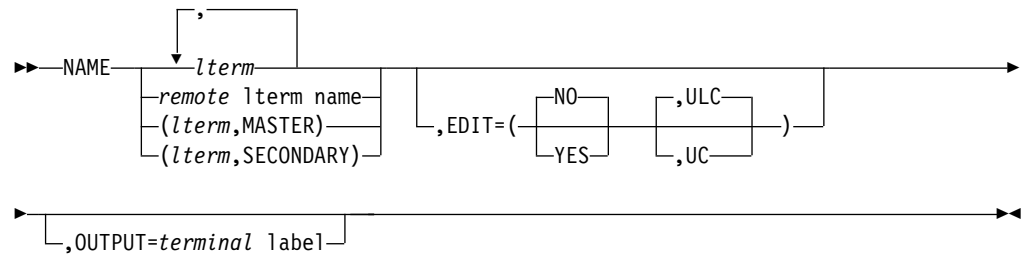
It can be desirable to use dynamic rather than fixed allocation for one parallel session (SESSION=1). This permits all subpools to be available for dynamic allocation to that session. This function is similar to SUBPOOL support offered for BTAM dial terminals.

Figure 5 summarizes the order of the macros:

Fixed LTERM Allocation Dynamic LTERM Subpool Allocation

TYPE	UNTYPE=LUTYPE6	UNTYPE=LUTYPE6
TERMINAL	SESSION=1	TERMINAL SESSION=(1-255)
NAME	1trmname	.
		.
		.
		VTAMPOOL
		SUBPOOL NAME=SBPLNAME
		NAME 1trmname

Figure 5. Order of Macro Statements for Fixed and Dynamically Allocated Subpools



Master Terminal



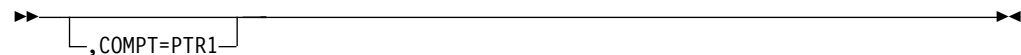
3601 Work Station



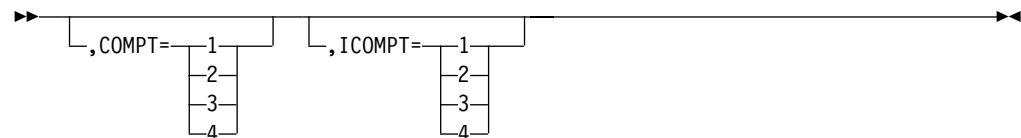
2780 Terminal



3275 Terminal



SLUTYPE1, SLUTYPEP, LUTYPE6 Terminals



Positional Parameters

lterm or lterms

Specifies one or more 1- to 8-character names for logical terminals associated with previously defined physical terminals. The name can contain only alphanumeric characters, that is, the letters A through Z, digits 0 through 9, and

NAME Macro

national characters # \$, and @. WTOR and DFSMTCNT are invalid *lterm* names. The entered name cannot begin with the character sequence INQU. Logical terminal names, transaction codes, and MSNAME link names, collectively, cannot contain duplicates. This operand is required. For further restrictions, refer to "Resource Naming Rules" on page 54.

If you enter the operand as a sublist with the keyword MASTER as a second parameter, the entered name is the identifier of the IMS master terminal.

If you enter the operand as a sublist with the keyword SECONDARY as the second parameter, the entered name is the identifier of the IMS secondary master terminal.

Examples of multiple specifications are:

```
NAME LTERMA,LTERMX,LTERMZ
NAME LTERMA,(LTERMX,MASTER),LTERMZ
```

One logical terminal in the system must be specified as the master terminal. The logical terminal chosen as the master terminal cannot be on a switched LINEGRP. It must be one of the following types:

- 2740 with station control
- 3270 display terminal
- Terminal defined as either SLUTYPE1 or SLUTYPE2

It must also be the console component if the terminal is defined as SLUTYPE1.

One logical terminal can be specified as the secondary master terminal. This logical terminal cannot be on a switched line group. It must be one of the following types:

- A 2740 with station control
- A 328x physical terminal
- A terminal defined as SLUTYPE1
- A terminal in a line group defined as a SPOOL

If a 2740 is chosen as the secondary master terminal, the logical terminal chosen as the primary master terminal and the logical terminal chosen as the secondary master terminal must be associated with the same 2740 physical terminal.

For terminals with multiple NAME LTERMs assigned, system error message responses are sent to the logical terminal name that is lowest in the collating sequence, unless the terminal is an input-only terminal. For input-only terminals, the system responses go to the first assigned logical terminal OUTPUT= terminal.

If multiple NAME LTERMs are assigned and an LTERM other than the first is specified as MASTER or SECONDARY, data and commands are not designated by the system as having originated at a master terminal.

remote lterm name

Specifies a one- to eight-character name for a logical terminal associated with a physical terminal defined in a remote IMS system. The name can contain only alphanumeric characters. WTOR and DFSMTCNT are invalid logical terminal names. The name cannot begin with the character sequence INQU. For further

restrictions, refer to "Resource Naming Rules" on page 54. No other operands are meaningful for a remote logical terminal; if provided, they are interpreted as comments only.

Keyword Parameters

To find which parameters apply to your IMS configuration refer to Table 12 on page 32.

COMPT=

Specifies the output component associated with this logical terminal. If this is the master terminal, the COMPT= operand value must be either not specified or 1.

For 3601 work stations, you can specify 1, 2, 3, or 4.

For 2780 terminal components, you can specify 1 or PTR1, or 2 or PCH1.

For 3275 terminals, you can specify only PTR1. PTR1 refers to a 3284 Model 3 printer attached to a 3275 terminal, within a remote 3270 line group or switched 3275.

COMPT= cannot be specified for NTO devices.

EDIT=

Specifies whether the logical terminal user-supplied edit routine DFSCNTE0 is to be used when routing a message to this logical terminal. The logical terminal edit routine is only included within the IMS control program nucleus if one of the NAME statements specifies EDIT=YES. If ULC is specified, output is transmitted as received. If UC is specified, output is translated to uppercase before transmission.

The logical terminal edit routine DFSCNTE0 is not supported for ETO logical terminals.

ICOMPT=

Specifies the input component associated with the terminal defined in the previous TERMINAL macro. If this is the master terminal, the ICOMPT= operand value must be either not specified or 1.

For SLU 1, SLU P, or LU 6.1 terminal systems, you can specify 1, 2, 3, or 4. These values relate to the components specified in the preceding TERMINAL macro statement.

For secondary logical units defined as SLUTYPE1, if the first component is a console, the default is 1; if the first component is a printer, no default exists. For terminals defined as SLUTYPEP or LUTYPE6, the default is 1.

ICOMPT= cannot be specified for NTO devices.

OUTPUT=

Specifies a terminal that is to be used as the output terminal for this logical terminal name. This is specified by entering the label of the TERMINAL macro defining the desired terminal. The referenced TERMINAL statement must occur before this NAME statement. OUTPUT= cannot be specified for VTAM terminals. OUTPUT= cannot be used with response mode, Fast Path, or conversational transactions.

transactions that do not cause a change to a database. Programs are prohibited from issuing Insert, Delete, or Replace calls to a database when processing a transaction defined as INQ=YES.

An APPLCTN macro statement that defines a message-driven Fast Path application program should be followed by at least a TRANSACT macro statement or a RTCODE statement.

SECURITY Macro

The SECURITY macro statement lets you specify optional security features to be in effect during IMS execution unless they are overridden during system initialization.

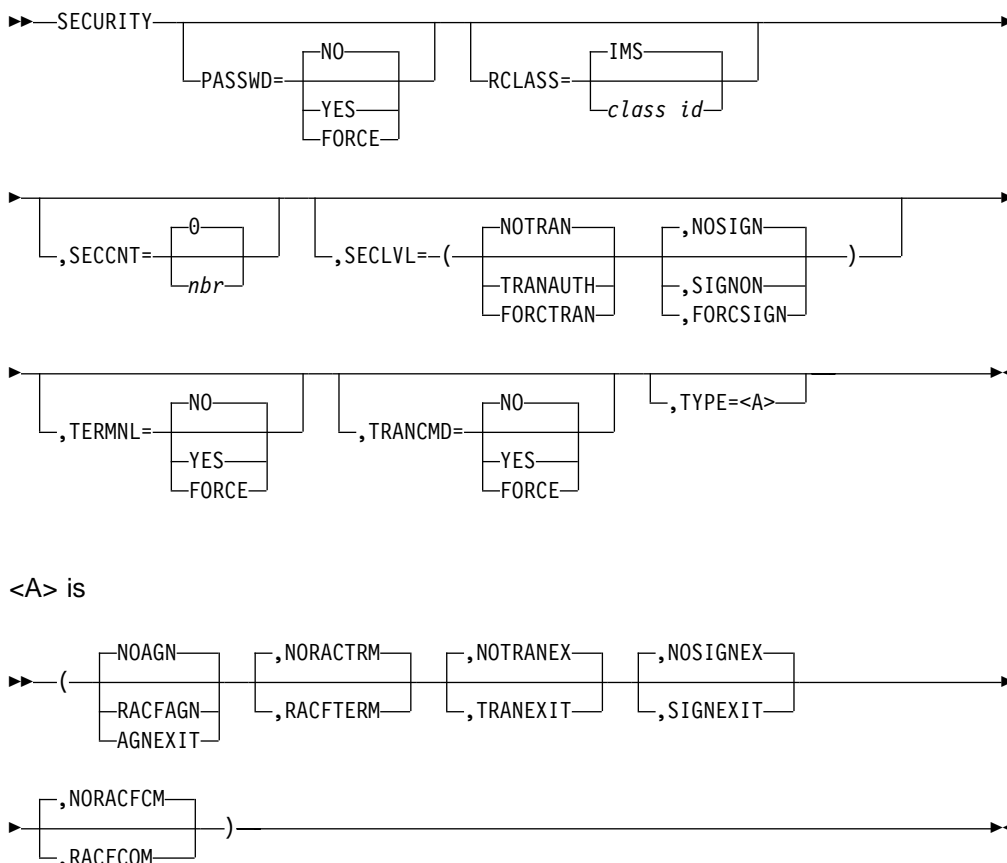
The IMS system can be defined to use the Resource Access Control Facility (RACF) licensed program (or equivalent), an exit routine, or both to perform transaction authorization, command authorization, or signon verification.

RACF or an exit routine (but not both) can be used to establish PSB authorization verification and application resource, or DBCTL, access authority checking. Authority to issue specific IMS commands can be restricted to certain transactions, and password and terminal security requirements can be imposed.

Related Reading: For full details of the security options and their use, see “Establishing IMS Security” in *IMS/ESA Administration Guide: System*.

If the SECURITY macro is included, any values specified or accepted by default override options in the COMM or MSGEN macro.

SECURITY Macro



Keyword Parameters

To find which parameters apply to your IMS configuration, for example to DBCTL, refer to Table 12 on page 32.

PASSWD=

Specifies password security enforcement level options available to the MTO when restarting IMS.

NO | YES | FORCE

Specifies whether (FORCE or YES) or not (NO) password security tables are to be loaded from the MATRIX data set during IMS restart.

If YES or NO is specified, this option can be overridden by the MTO during restart. The default is NO.

RCLASS=

Specifies an identifier, one to seven alphanumeric characters, used to identify the IMS system as a resource class to RACF for transaction authorization and user ID verification.

The specification is valid only if TYPE=RACFTERM or RACFAGN is specified. Application resource access authorization uses the IMSID specification on the IMSCTRL macro statement.

IMS class names do not need to be unique, but if you do make them unique you can, for example, define the same transaction name on a production subsystem and a test subsystem, with different access lists for each subsystem, with no ambiguity.

The default is IMS.

SECCNT=

Specifies the maximum number of terminal and password security violations to be accepted per physical terminal and the number of transaction command violations per transaction prior to master terminal notification of such violations. The default is 0, which nullifies notification to the master terminal. The number specified must be 0, 1, 2, or 3.

If SECCNT is not 0, the master terminal is notified for every violation.

SECLVL=

Specifies the user ID verification and transaction authorization security enforcement level options available to the MTO when restarting IMS.

Related Reading: For further information on the effect of these options and the relationship between these operands, JCL, and the /NRE command, refer to *IMS/ESA Administration Guide: System*.

NOTRAN | TRANAUTH | FORCTRAN

Specifies whether (FORCTRAN or TRANAUTH) or not (NOTRAN) transaction authorization is to be performed by the online IMS system.

If TRANAUTH or NOTRAN is specified, this option can be overridden by the MTO at restart. The default is NOTRAN.

A specification of FORCTRAN or TRANAUTH requires that TYPE=RACFTERM or TRANEXIT, so that the processing can be performed.

If FORCTRAN or TRANAUTH is specified, the requirements for user ID verification must also be considered.

Related Reading: For further details on the effect of these options, refer to *IMS/ESA Administration Guide: System*.

NOSIGN | SIGNON | FORCSIGN

Specifies whether (FORCSIGN or SIGNON) or not (NOSIGN) /SIGN ON user ID verification security tables are to be loaded from the MATRIX data set during IMS restart.

If SIGNON or NOSIGN is specified, this option can be overridden by the MTO at restart. Specification of FORCSIGN or SIGNON requires that TYPE=RACFTERM or SIGNEXIT, so that processing can be performed.

Transaction authorization requires a validated user ID in order to verify that the user is authorized to invoke the transaction. For this reason, the security level of user ID verification must be at least as great as that of transaction authorization. The requirements are shown in the following table:

Parameter	Default	Invalid
NOTRAN ⁴	NOSIGN	- -

SECURITY Macro

TRANAUTH	SIGNON	NOSIGN
FORCTRAN	FORCSIGN	SIGNON
FORCTRAN	FORCSIGN	NOSIGN

TERMNL=

Specifies terminal security enforcement level options available to the MTO when restarting IMS.

NO | YES | FORCE

Specifies whether (FORCE or YES) or not (NO) terminal security tables are to be loaded from the MATRIX data set during IMS restart.

If YES or NO is specified, this option can be overridden by the MTO during restart. The default is NO.

Related Reading: For further details on the effect of these options, refer to *IMS/ESA Administration Guide: System*.

TRANCMD=

Specifies transaction command security enforcement level options available to the MTO when restarting IMS.

NO | YES | FORCE

Specifies whether (FORCE or YES) or not (NO) transaction command security tables (if any) are to be loaded from the MATRIX data set during IMS restart.

If YES or NO is specified, this option can be overridden by the MTO during restart by specifying the NOTRANCMD or TRANCMD keyword, respectively, on the /NRESTART command. The default is NO.

Related Reading: For further details, refer to *IMS/ESA Administration Guide: System*.

TYPE=

Specifies whether application resource access authorization, transaction authorization, or /SIGNON user ID verification are to be performed.

NOAGN | RACFAGN | AGNEXIT

Specifies whether (RACFAGN or AGNEXIT) or not (NOAGN) application resource access authorization is to be performed by IMS at execution time. Application resource access security checking can be performed by the RACF licensed program (RACFAGN) or by an exit routine named DFSISIS0 (AGNEXIT). The default is NOAGN.

NORACTRM | RACFTERM

Specifies whether (RACFTERM) or not (NORACTRM) linkage to the RACF program product is to be provided for transaction authorization or /SIGN ON user ID verification. If RACFTERM is specified, RACF is invoked whenever either of the two functions is required. The default is NORACTRM.

NOTRANEX | TRANEXIT

Specifies whether (TRANEXIT) or not (NOTRANEX) the user transaction authorization exit (DFSCTRN0) will be called by IMS to process transaction authorization requests at execution time. If both RACFTERM and TRANEXIT are specified, RACF is called first. The default is NOTRANEX.

4. Specified or default.

NOSIGNEX | SIGNEXIT

Specifies whether (SIGNEXIT) or not (NOSIGNEX) a user /SIGN ON exit (DFSCSGN0) is to be called by IMS to validate a user ID when entered at execution time. If both RACFTERM and SIGNEXIT are specified, RACF is called first. If TYPE=SIGNEXIT is specified in the SECURITY macro, SECLVL=SIGNON is also required to enforce user ID verification.

The default is NOSIGNEX; however, SIGNEXIT is assumed and forced if TRANEXIT is specified and NORACTRM is specified or accepted by default, because user ID verification is a prerequisite for transaction authorization checking.

NORACFCM | RACFCOM

Specifies whether (RACFCOM) or not (NORACFCM) RACF is to be called to verify command authorization. This applies to commands from ETO terminals. The default is NORACFCM.

Relationships between the SECURITY, COMM, and IMSGEN Macros

The SECURITY macro is provided to enhance IMS system security. It can replace the security specifications on the IMSGEN and COMM macros, both explained elsewhere in this manual. This section puts into perspective the relationship that exists between these three macros and suggests a possible installation strategy.

All security specifications have now been consolidated in the SECURITY macro. However, for compatibility with previous releases, the COMM or IMSGEN keywords related to security specifications can still be used.

The security-related keywords from these three macros are accepted hierarchically in the order SECURITY, COMM, and then IMSGEN; that is, if the SECURITY macro is used, the specifications and defaults from that macro will take precedence over any security specifications coded on either COMM or IMSGEN.

If security specifications are coded on more than one of these macros, a warning message is issued, indicating that the specifications and defaults from the highest macro in the hierarchy present in the stage 1 input is used. Likewise, if the COMM macro is used and any of the IMSGEN macro's general communication options are specified, a warning message is issued indicating that the general communication specifications and defaults from the COMM macro are used.

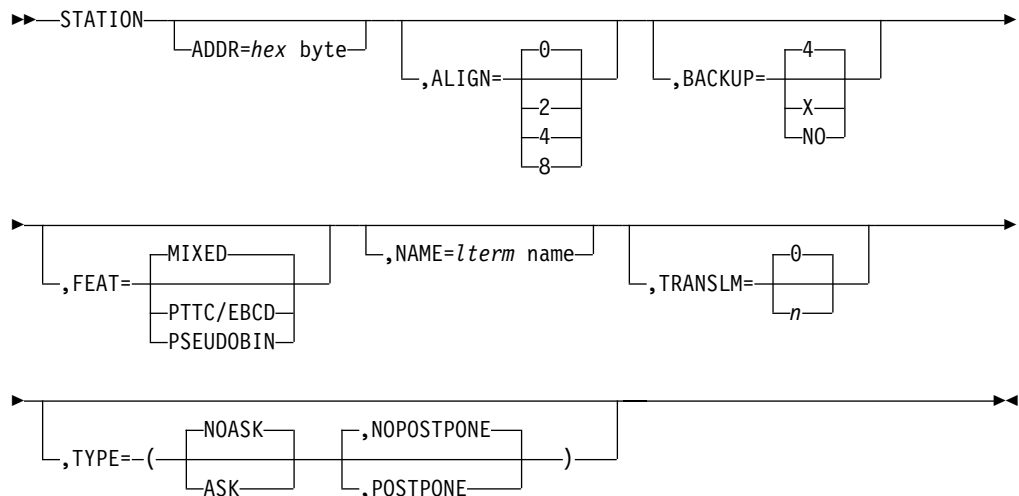
SPAREA Macro

The SPAREA macro is no longer supported because DASD SPAs are no longer supported. If you specify the SPAREA macro, it is ignored.

STATION Macro

The STATION macro statement describes the physical and logical characteristics of the System/3 or System/7.

STATION Macro



Keyword Parameters

To find which parameters apply to your IMS configuration refer to Table 12 on page 32.

ADDR=

Specifies the two-digit hexadecimal representation of the System/3 or System/7 address. This operand is not required for System/3 or System/7 contention line groups.

If either of the following conditions exists on the stage 1 input macros for a System/3 or a System/7 BSC line group, the address coded on the TERMINAL macros following the STATION macro must not be of the form XYXY, where X and Y are any hexadecimal digits:

- The address (ADDR=) is not coded on the STATION macro, and the coded address on the preceding LINE macro is of the form 0XY.
- The coded address on the STATION macro is of the form XY.

ALIGN=

Specifies that each data segment is to be right justified or padded to halfword, fullword, or doubleword (2, 4, or 8, respectively) alignment. 0 specifies that no alignment is to be performed. The default is 0.

BACKUP=

Specifies (for XRF only) the control of the automatic terminal switching after takeover. Use only when HSB=YES is specified on the IMSCTRL macro.

X is a numeric integer from 1 to 7, inclusive, that sets priority for reestablishing the session. It defaults to 4 when either the keyword or the parameter is omitted. Specifying BACKUP=NO suppresses the session recovery of the terminal at takeover.

FEAT=

Specifies whether data blocks are to be transmitted in PTTC/EBCD (PTTC/EBCD) code, pseudobinary PTTC/EBCD (PSEUDOBIN) code, or either (MIXED). This operand is valid for System/7 (START/STOP) line groups only. The default is MIXED.

NAME=

Specifies a logical terminal name for the station CNT. The default is a system-assigned sequence number with the prefix RSTS.

TRANSLM=

Specifies the maximum number of blocks to be transmitted to the System/3 or System/7 for each output request. This operand must be specified as a decimal number from 0 through 15. A specification of 0 indicates no limit. The default is 0.

TYPE=

Specifies whether output should be allowed to queue until such time as the station asks for the output (ASK), or whether output should be sent to the station as it becomes available (NOASK). The default is NOASK.

The second parameter specifies that this station should not receive output (POSTPONE) or should receive output (NOPOSTPONE) when the station is started. The default is NOPOSTPONE.

SUBPOOL Macro

For Switched Communication Devices: The SUBPOOL macro statement, when used within a switched communication device macro set, defines a set of logical terminals. This set is within a pool of logical terminals. The set is associated with a physical terminal on a switched communication line when the /IAM command is executed. Subpools can be defined within a POOL macro statement. At least one subpool must be defined for each POOL macro statement.

When used within a switched communication device macro set, the SUBPOOL macro statement has no operands. It must be followed by one or more NAME macro statements. These NAME statements can contain the COMPT= operand. If the COMPT= operand is present on any NAME statement within a subpool, it must be defined for a switched 3275. If the COMPT= operand is not present on any NAME statement within a SUBPOOL, it can be defined within a 2740 line group or for a switched 3275.

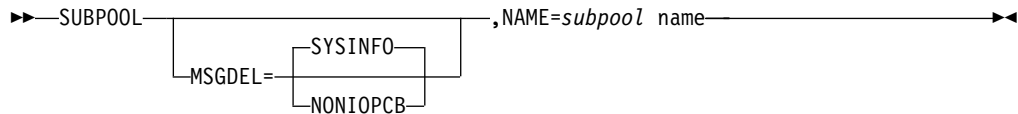
Related Reading: Refer to “Switched Communication Network” in the “Logical Terminal Network Design” section of *IMS/ESA Administration Guide: System* for a discussion on the use of pools and subpools.

For LU 6.1 VTAM Devices: The SUBPOOL macro statement, when used in a VTAM macro set, is a delimiter between groups of NAME macro statements to create LU 6.1 LTERM subpools. Although at least one NAME macro statement must be defined within the VTAMPOOL, it is valid to define one or more subpools with no NAME statements. This definition creates a reserved subpool to which no terminals are allocated until reassigned there by use of the /ASSIGN command.

The subpools must be named to be used in IMS commands or session initiation requests. The LU 6.1 subpool names, specified on the required NAME keyword operand of the SUBPOOL macro statement, must follow standard MVS naming conventions and must be unique with respect to other LTERM subpool names within the VTAMPOOL. The subpool names must be unique among themselves, but need not be unique to other names used in an IMS system, such as LTERM names and transaction names. These subpools can only be used with statically defined ISC terminals. They cannot be used with ETO terminals.

SUBPOOL Macro

If parallel sessions are defined in your system, at least one subpool must be defined. It is recommended that at least one subpool be defined for each parallel session defined in your system. Additional subpools can be defined if you desire. For a session to be brought up, a subpool must be allocated to it. Thus, if you have defined two parallel sessions in your system and only one subpool, only one session can be brought up at any one time.



Keyword Parameters

To find which parameters apply to your IMS configuration refer to Table 12 on page 32.

MSGDEL=

Used for VTAM LU 6.1 devices and specifies which message types IMS should discard for this terminal. The default is SYSINFO.

SYSINFO | **NONIOPCB**

SYSINFO, the default, specifies that IMS should discard DFS059 TERMINAL status messages for this terminal and also DFS3650 for SLU P devices.

NONIOPCB specifies that IMS should discard the following message types destined for this terminal:

- Message switches
- Messages inserted by an application program to an alternate PCB
- /BROADCAST messages
- DFS059 TERMINAL status messages

NAME=

Used for VTAM LU 6.1 devices and names the LU 6 subpool. The NAME keyword parameter is not allowed for BTAM subpools.

TERMINAL Macro

The TERMINAL macro statement defines physical and logical characteristics of VTAM nodes and non-VTAM communication terminals.

The NAME macro statements that follow a TERMINAL macro statement supply the logical terminal names that are associated with the physical terminal at system definition. Whichever terminal name is designated in the first NAME macro statement that follows a TERMINAL macro statement becomes the response or input/output logical terminal.

A TERMINAL macro statement that defines terminals connected to a switched communication line must not be immediately followed by a NAME macro statement. See 139 for the order in which the NAME macro statement appears in this instance.

All Fast Path-eligible terminals must operate in response (forced or transaction) mode. In addition, RESP=TERM must be specified on the LINE macro statement

for Fast Path-eligible 2740 terminals. Where applicable, the PAGDEL option must be specified (or used by default) for these terminals.

Message Format Service (MFS) can be used in an IMS Fast Path configuration.

The TERMINAL macro statement can be specified without operands if the terminal is a non-station control 2740-1, 2741, printer, punch, tape, or disk.

All non-VTAM data communication specifications must precede the VTAM macro set in your IMS system definition stage 1 input deck. You receive a stage 1 output warning message if the VTAM macro set is not the last physical set. If an MSC macro set is part of your system definition, it must precede the VTAM macro set, or your system definition does not complete. To add VTAM support, you must specify either an ON-LINE or ALL system definition on the IMSCTRL macro statement.

For VTAM terminals, the default values shown for operands of the TERMINAL macro statement are ignored if these operands are specified on the TYPE macro statement.

If system definition statements from previous releases of IMS are being used, TERMINAL macro definitions from the previous releases are valid for this release. The TERMINAL keywords in this section must be used for terminals newly supported with this release of IMS and for terminals that use new IMS terminal functions.

The TERMINAL macro statement can be used to define secondary logical units type 1, type 2, type 4, and type P, logical unit type 6.1, and NTO devices. These are designated on the TYPE macro as UNITYPE=SLUTYPE1, SLUTYPE2, SLUTYPEP, LUTYPE6, and NTO, respectively.

By specifying SLUTYPE1 on the TYPE macro, you can define an appropriate configuration of console, bulk printer, disk and card reader/punch. The 3767 and 3770 nonprogrammable terminals must be defined to IMS as SLUTYPE1.

SLUTYPE2 must also be specified for display devices attached to a 3274 or 3276 Control Unit that is operating in SNA mode.

Related Reading: For 3270/SLUTYPE2 dynamic terminals that are not defined by the TERMINAL macro, see MFS Device Characteristics Table Utility in *IMS/ESA Utilities Reference: Transaction Manager*.

By specifying SLUTYPEP on the TYPE macro, you can define 3600 terminals and 3790 or later programs using the Host Communication Facility. This support extends full IMS functional capabilities to user-written programs within the controller and includes the MFS Distributed Presentation Management (DPM) function. The support is in addition to current support for 3600 and 3790 user-written programs.

By specifying LUTYPE6 on the TYPE macro, you can define a logical unit type 6 node such as CICS, another IMS, or a user-written program. This node can communicate with IMS as an SNA primary or secondary half-session.

For logical units type 6, the log write-ahead option, as specified by the LTWA and NLTWA parameters on the TERMINAL macro statement, need no longer be specified, because log write-ahead occurs automatically. Therefore, although this parameter can be retained for compatibility with previous releases, if it is specified, it is ignored by IMS.

TERMINAL Macro

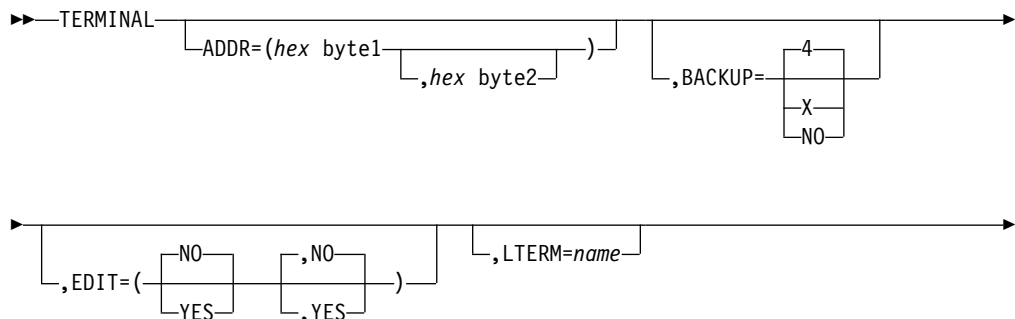
By specifying NTO on the TYPE macro, you can define NTO devices. NTO support provides a non-SNA start/stop terminal interface to VTAM for 3101, 2740-1, 2741, TTY, and TTY-compatible devices. The TERMINAL macro statement keyword PU= permits you to designate the type of terminal being defined to use NTO support.

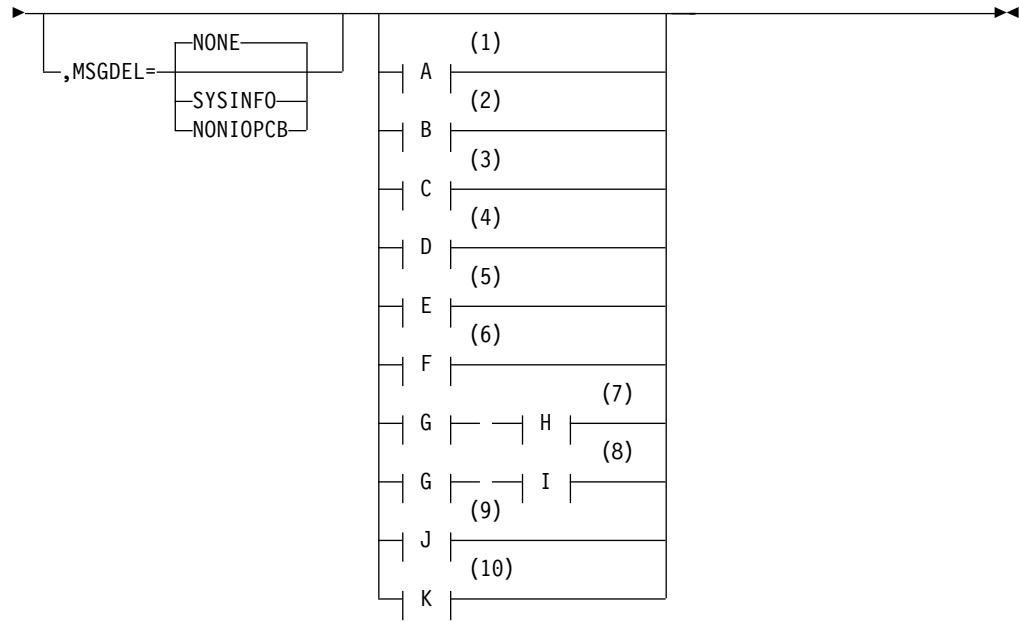
Related Reading: For further information on SLU P and LU 6.1 (ISC), see the *IMS/ESA Administration Guide: Transaction Manager*.

The TERMINAL macro statement can also be used to define terminals belonging to a Finance Communication System such as the 3600 Finance Communication System or the 4700 Finance Communication System. Finance Communication System components for devices specified as UNITYPE=3601 or as UNITYPE=FINANCE on the TYPE macro statement can be defined on the TERMINAL macro statement using either specific 3600 definitions or generic FINANCE terminal definitions. The code generated to support the terminal is identical regardless of whether the terminal is specified as UNITYPE=3601 or UNITYPE=FINANCE.

By specifying SLUTYPE2 on the TYPE macro, you can define 3277 terminals, Models 1 and 2, operating under control of the 3790/3270 Data Stream Compatibility. Under this support, the 3277 Model 2 can be defined as the IMS master terminal; the Model 1 is not supported as an IMS master terminal. This support is in addition to current support for the 3600 and 3790 user-written programs. The 3790 can be defined appropriately as SLUTYPE1, SLUTYPE2 or as a subset of FINANCE.

All Non-VTAM Terminals

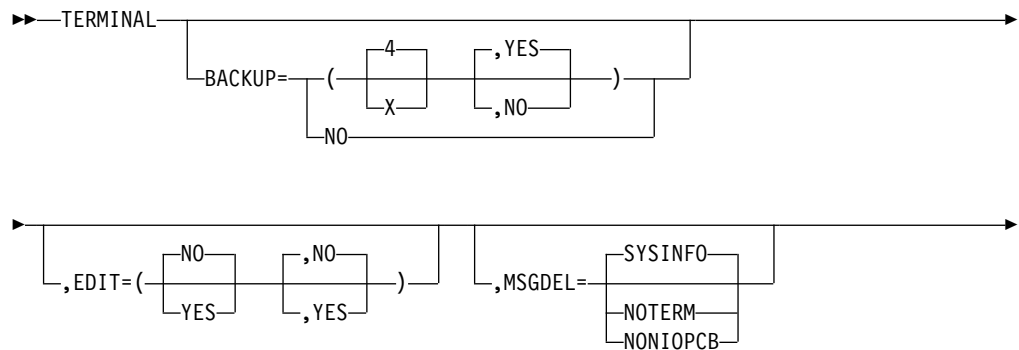




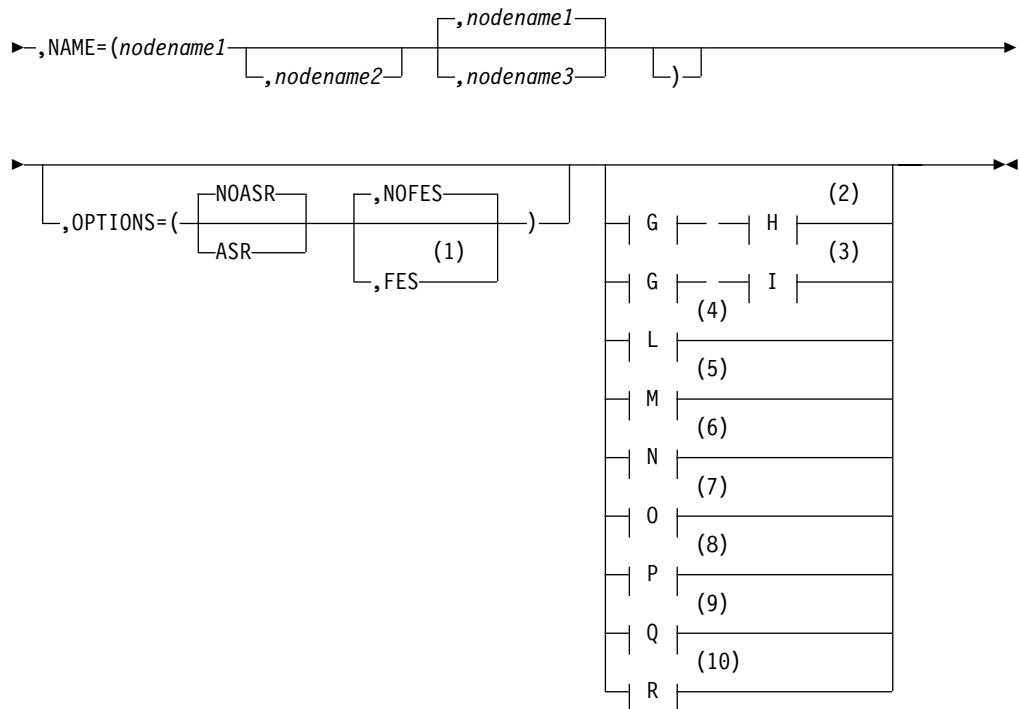
Notes:

- 1 2740 Non-Station Control Terminal
- 2 2740 Switched Terminal
- 3 2740 Station Control Terminal
- 4 2740 Model 2 Terminal
- 5 2741 Terminal
- 6 2780 Terminal
- 7 3270 Local Terminal
- 8 3270 Remote Terminal
- 9 Spool Terminal
- 10 System/3 and System/7 Terminal

All VTAM Terminals



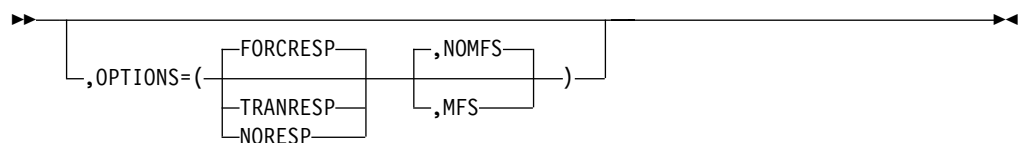
TERMINAL Macro



Notes:

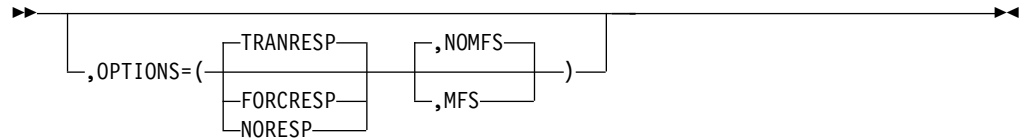
- 1 FES not valid for LU 6 terminals
- 2 3270 Local Terminal
- 3 3270 Remote Terminal
- 4 Finance Work Station (If UNITYPE=FINANCE on the TYPE macro)
- 5 3600 Work Station (If UNITYPE=3601 on the TYPE macro)
- 6 SLU 1 Terminal
- 7 NTO Device
- 8 SLU 2 Terminal
- 9 SLU P Terminal
- 10 LU 6 Terminal

A: (2740 Non-Station Control Terminal)

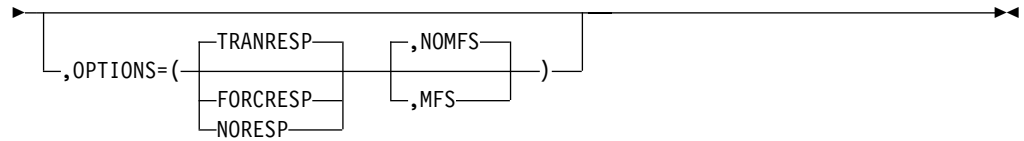


B: (2740 Switched Terminal)

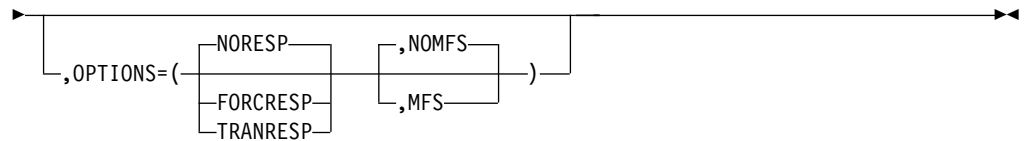
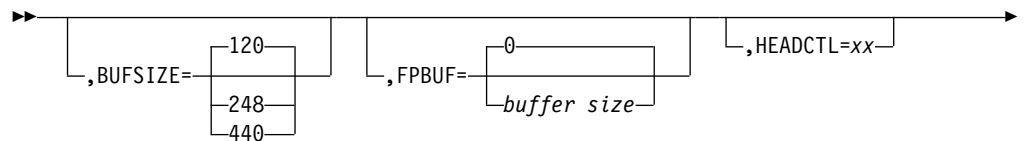
TERMINAL Macro



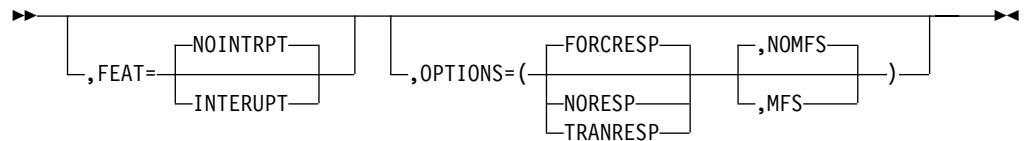
C: (2740 Station Control Terminal)



D: (2740 Model 2 Terminal)

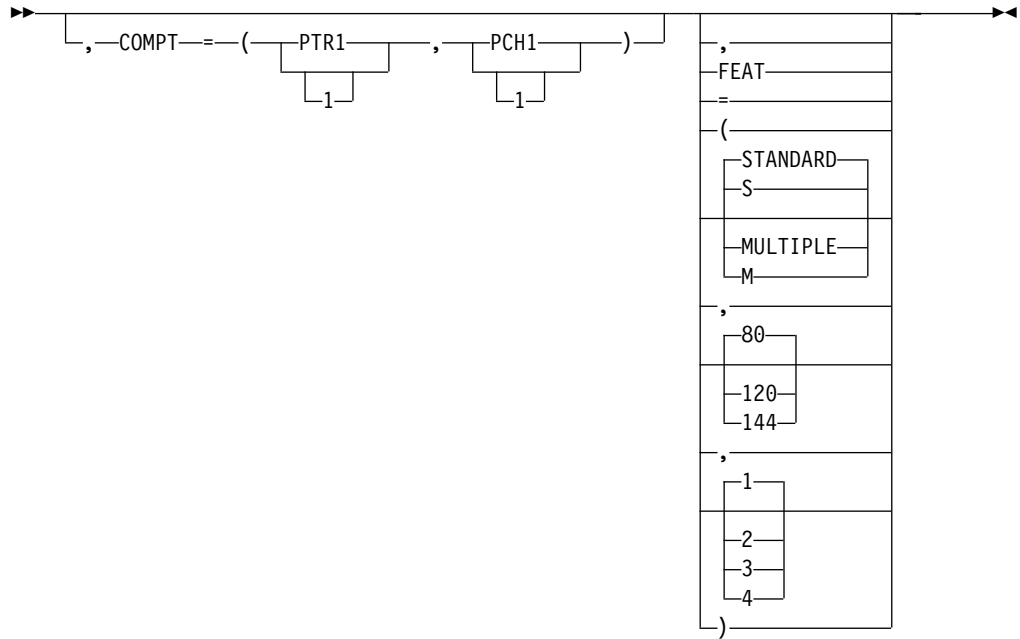


E: (2741 Terminal)

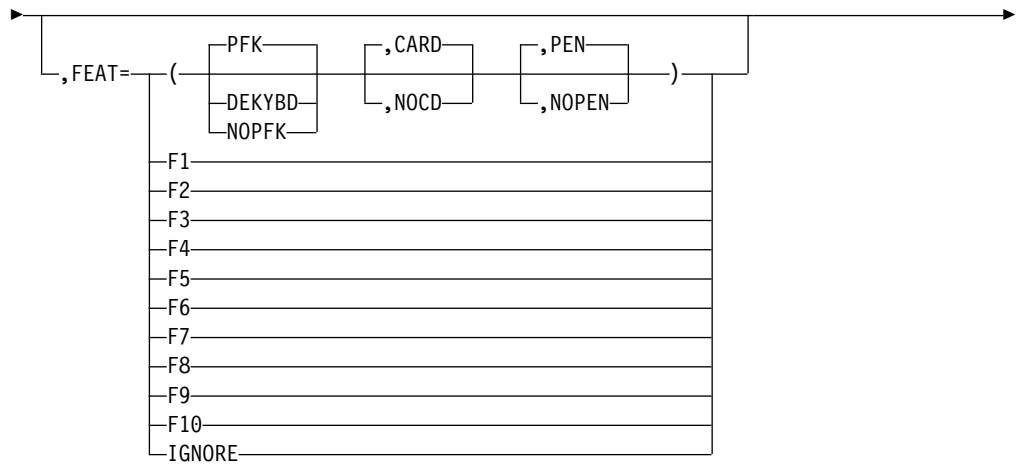
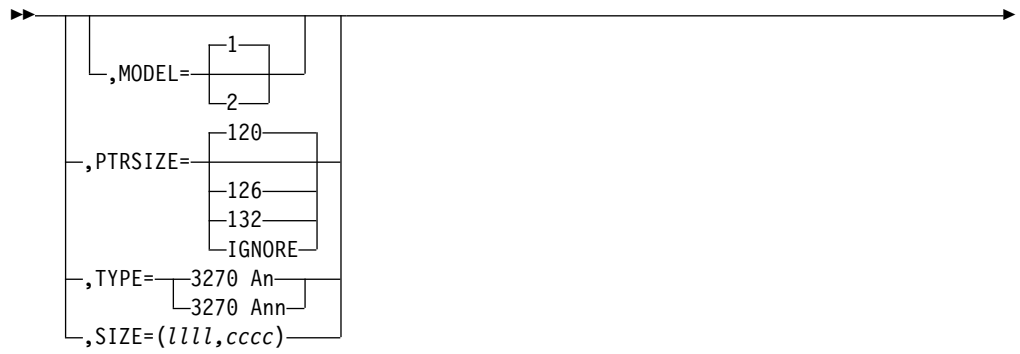


F: (2780 Terminal)

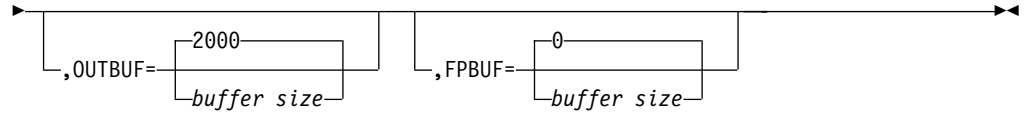
TERMINAL Macro



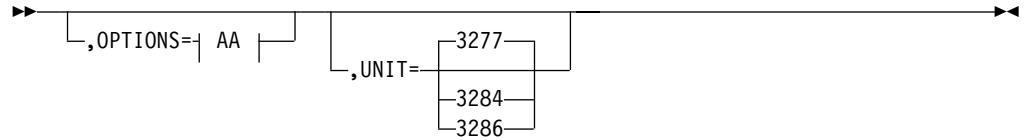
G: (All 3270 Terminals)



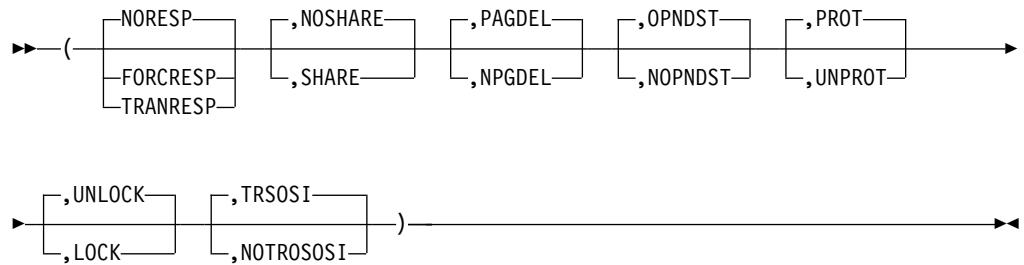
TERMINAL Macro



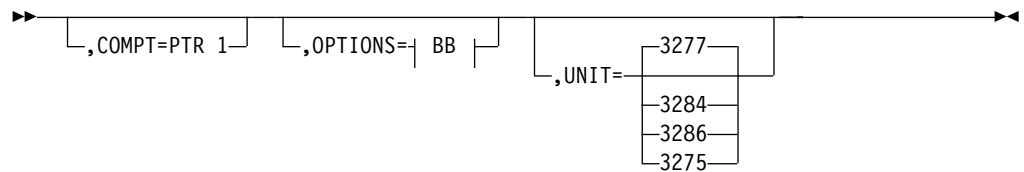
H: (3270 Local Terminals)



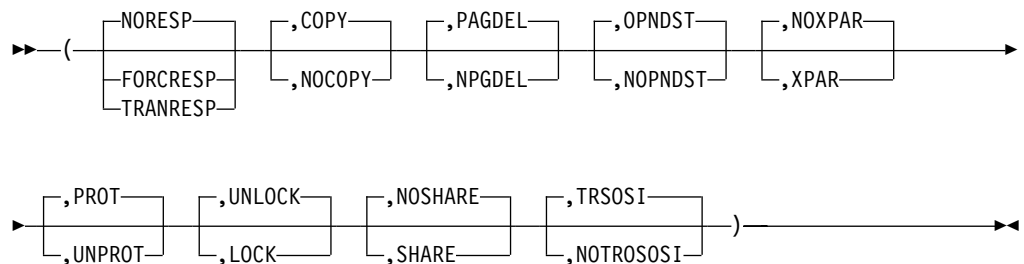
AA:



I: (3270 Remote Terminal)

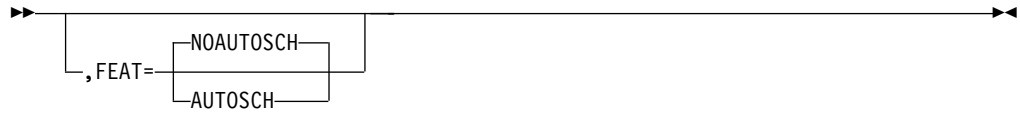


BB:



J: (Spool Terminal)

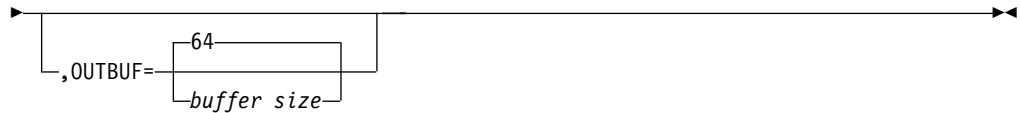
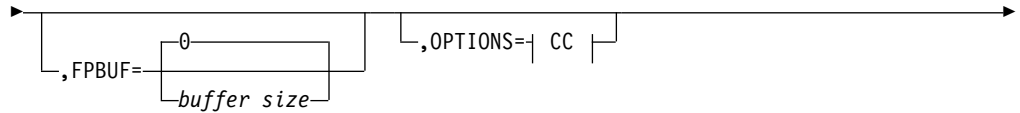
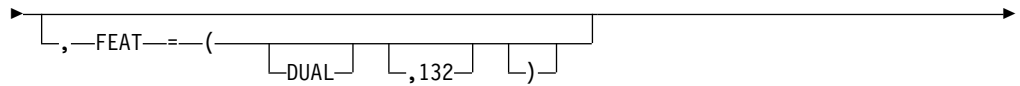
TERMINAL Macro



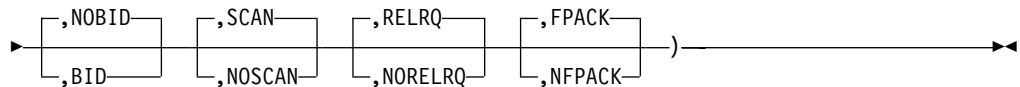
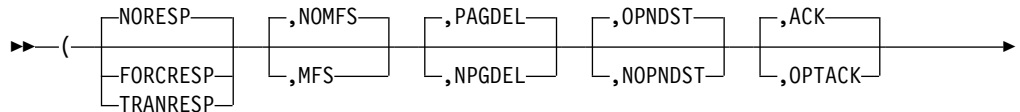
K: (System/3 and System/7 Terminal)



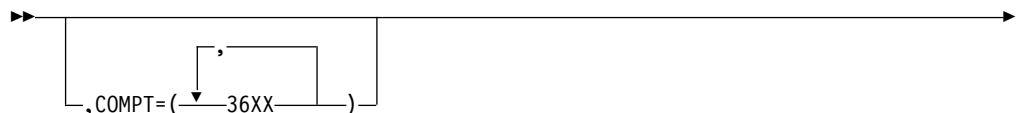
L: Finance Work Station (UNITYTYPE=FINANCE on the TYPE macro)



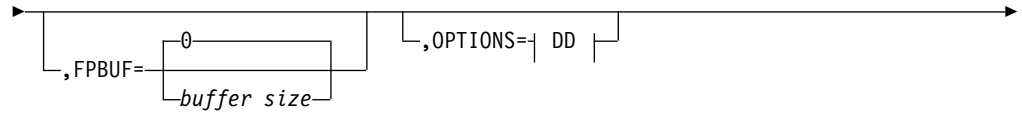
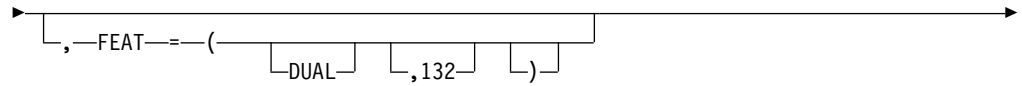
CC:



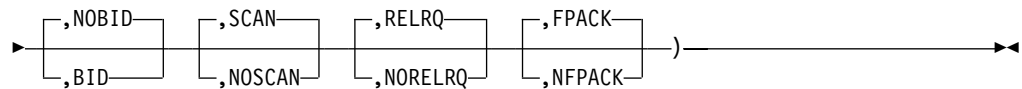
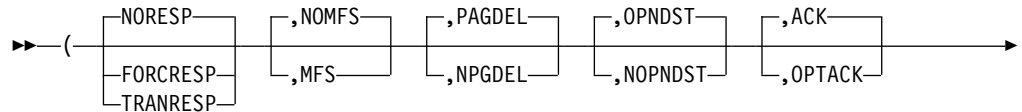
M: 3600 Work Station (If UNITYTYPE=3601 on the TYPE macro)



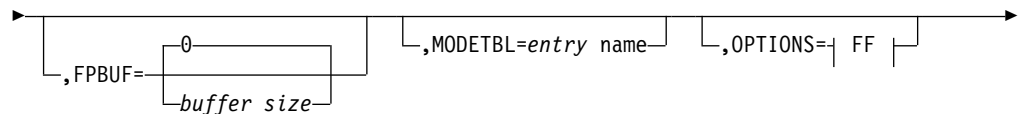
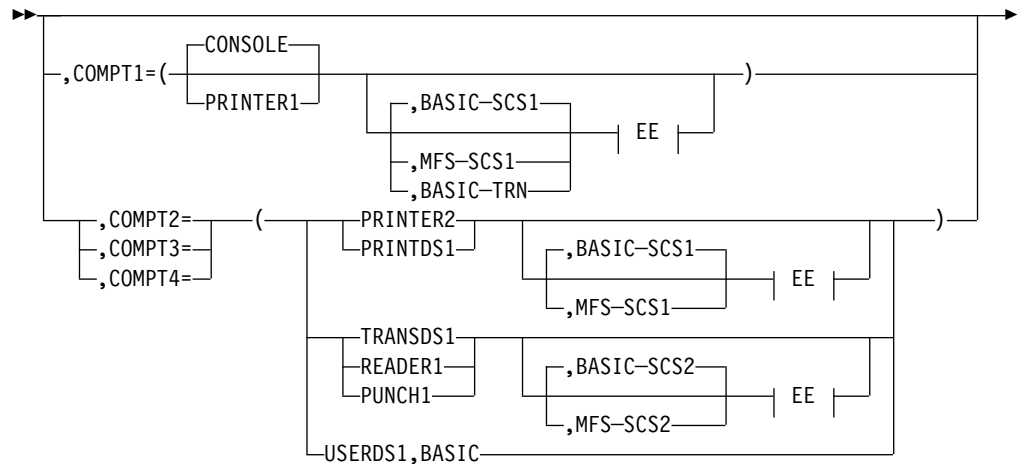
TERMINAL Macro



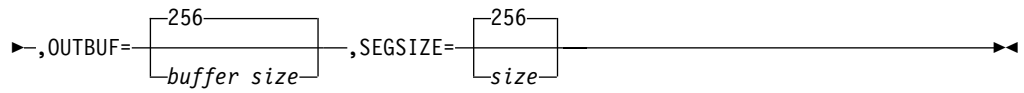
DD:



N: (SLU 1 Terminal)



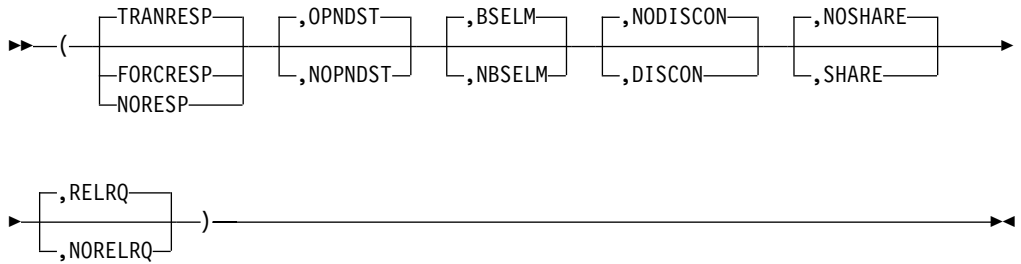
TERMINAL Macro



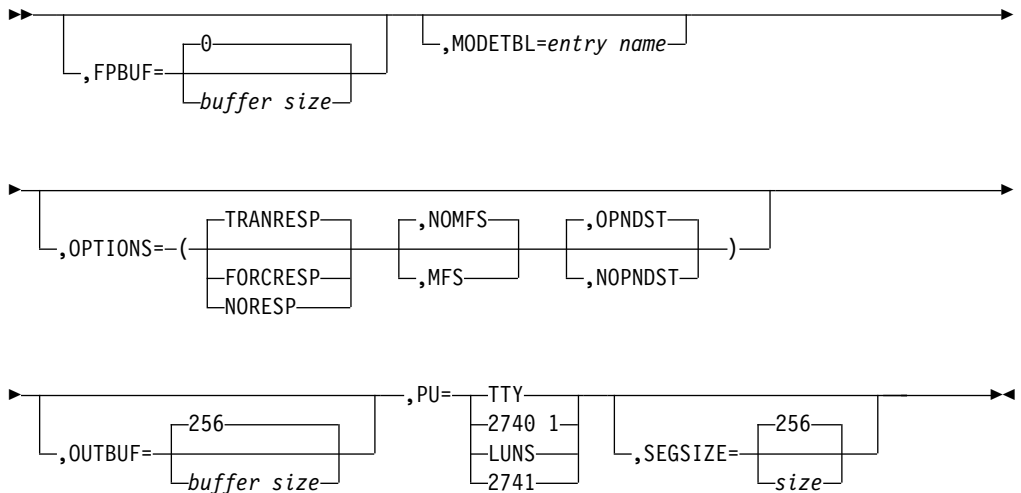
EE:



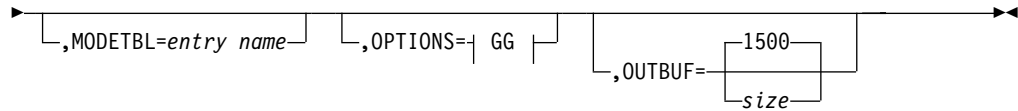
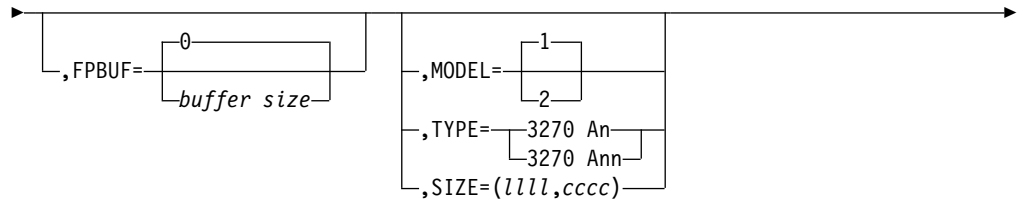
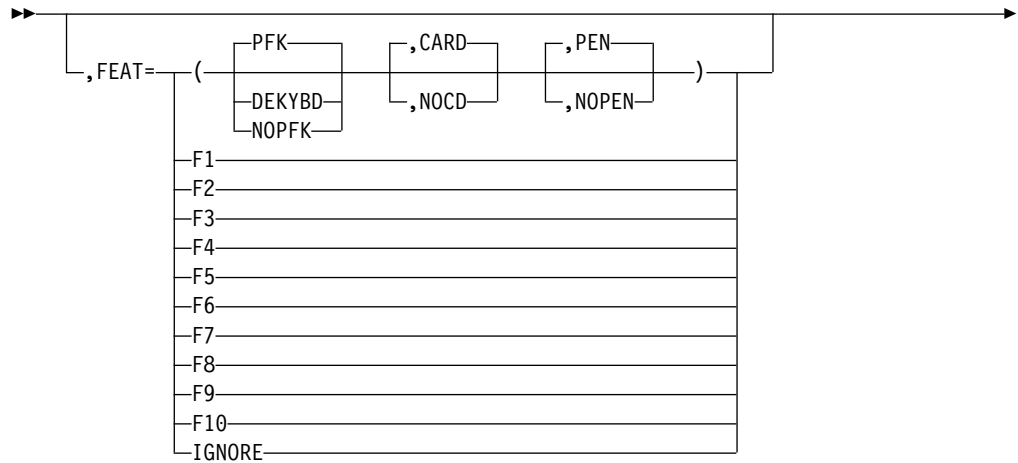
FF:



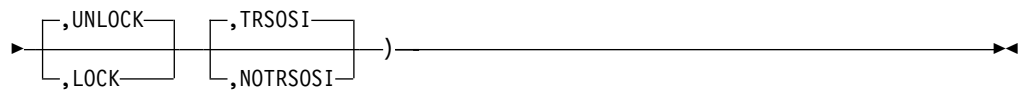
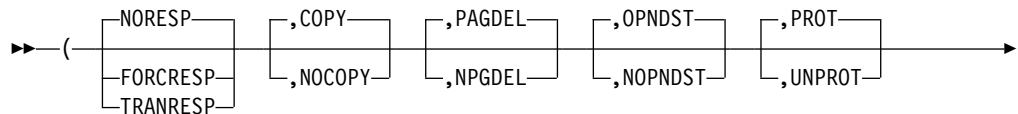
O: (NTO Device)



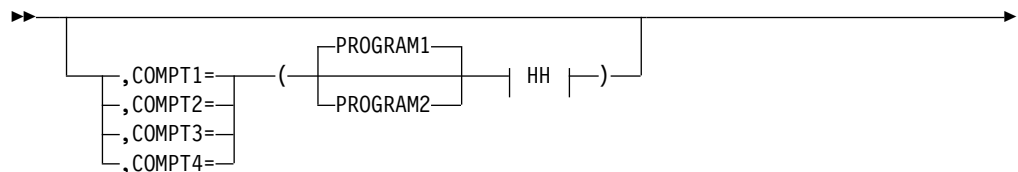
P: (SLU 2 Terminal)



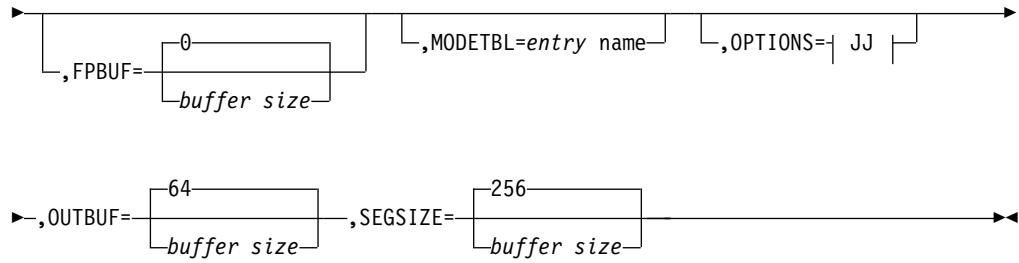
GG:



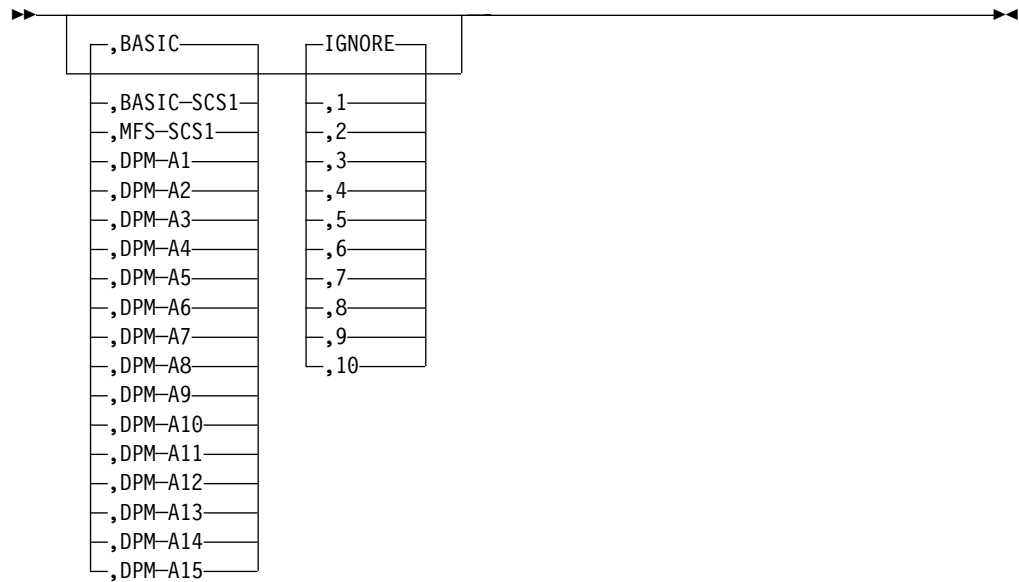
Q: (SLU P Terminal)



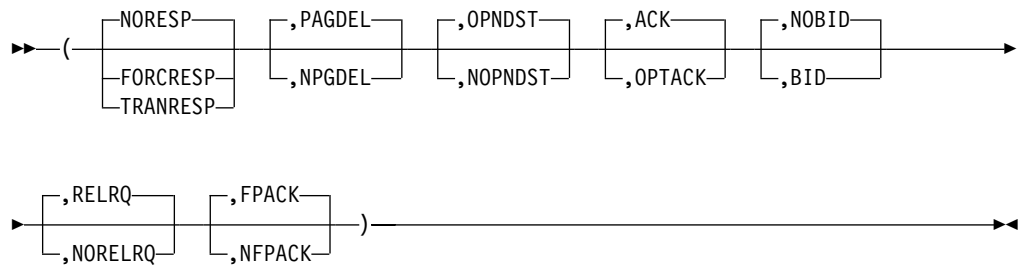
TERMINAL Macro



HH:

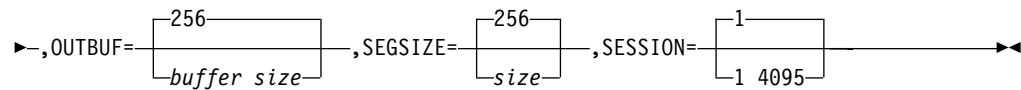
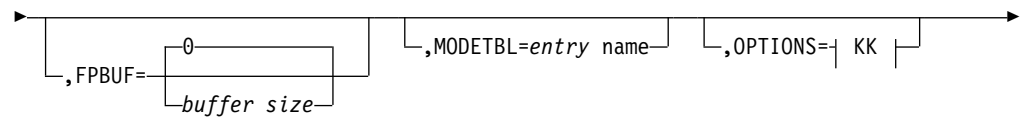
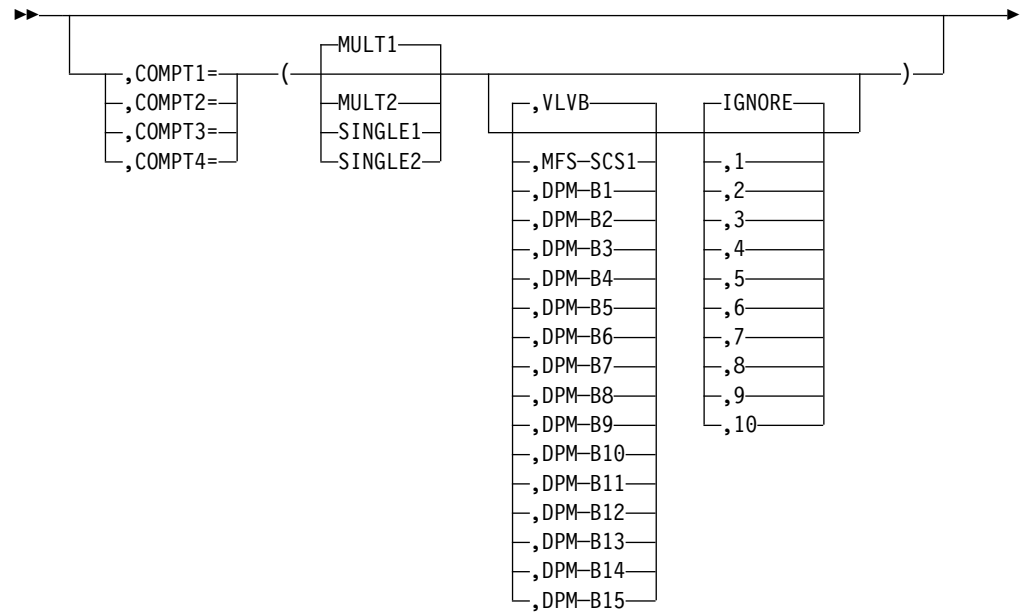


JJ:

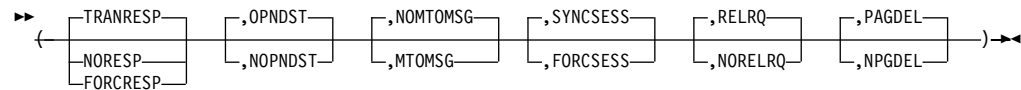


R: (LU 6.1 Terminal)

TERMINAL Macro



KK:



Label Field

if any subsequent NAME macro statement refers to this TERMINAL macro statement in its OUTPUT keyword, a label must be specified for the TERMINAL macro statement.

Keyword Parameters

To find which parameters apply to your IMS configuration refer to Table 12 on page 32.

ADDR=

Specifies, when XRF is not used, the physical terminal addressing character in terminal code (hexadecimal representation) for all terminals. For example, physical terminal address "A" for a 2740 is coded ADDR=E2 (E2 is 2740 line code for "A"). For specific terminal types, check the appropriate component description manuals for the types of addresses required.

TERMINAL Macro

If the terminal is a local 3270, the terminal address must be specified as three hexadecimal digits for MVS/ESA SP V4, or as three or four hexadecimal digits for MVS/ESA SP V5 or later.

If the terminal is a System/3, System/7, or nonswitched 3275, the address must be specified as four hexadecimal digits. For a nonswitched 3275 or 3276 terminal, the first two digits specify the control unit polling address, while the last two specify the terminal address (example, ADDR=4040).

If the terminal is a 3277 or 3284/3286 attached to a 3271, the address must be specified as two hexadecimal digits, specifying the terminal address related to the particular display station or printer.

Independent of terminal type, all input terminal addresses for a line must be unique. This operand is not required for 2741, switched 3275, READER, PRINTER, PUNCH, TAPE, DISK, SPOOL, or 2740 Model 1 non-station control terminals. For all other terminal types not previously mentioned, the address must be specified as two hexadecimal digits.

When XRF is used, a second set of polling characters should be specified in the ADDR= keyword. This additional parameter is only used to define the second address for a remote BTAM-supported IMS master or secondary master terminal. And, when specified, only the master terminal and the secondary master terminal can be in the line group. RSR does not support BTAM terminals.

BACKUP=

Specifies (for an XRF complex) control of the automatic restart or session switching (VTAM) after takeover. Use only when HSB=YES is specified on the IMSCTRL macro.

X is a numeric integer from 1 to 7, inclusive, that specifies the priority for reestablishing the session. The default is 4 if either the keyword or the parameter is omitted. NO suppresses session recovery of the terminal at takeover. The priority specified is effective not only in switching and reestablishing a session during takeover, but also for all session activities performed under the terminal.

Although requests to VTAM are prioritized by IMS, the active requests can be completed in any order because of internal VTAM conflicts and pacing.

The second parameter is only used with VTAM terminals. It determines whether the backup system is to attempt to establish a backup session to the terminal when the active request establishes a session. Backup sessions are not valid for UNITYPEs 3270, LU6, and NTO. If this parameter is specified as YES for these UNITYPEs, it is ignored. If it is not specified, the default is NO. The default for this second parameter is YES when either the keyword or the parameter is omitted. Specifying NO for this parameter suppresses the establishment of a backup session.

BUFSIZE=

For 2740 Model 2 lines, specifies the size of the output buffer for this 2740 Model 2 terminal. The value specified here cannot exceed the value specified on the preceding LINE macro statement. The default is 120.

COMPT=

Specifies components of this terminal. This operand is valid only for 2780, nonswitched 3275, and 3600 terminals.

- For 2780 terminals:
Valid entries are PTR1 and PCH1, or 1 and 2 respectively. These entries can appear in any sequence. If this keyword specification is omitted, it is assumed that the user has a full complement of output components for the device.
- For nonswitched 3275 terminals:
The only valid entry is PTR1. PTR1 specifies that a 3284 Model 3 printer is attached to this 3275 terminal. The video display is considered as component 1 and the 3284 Model 3 printer is considered as component 2.
- For terminals specified as UNITYPE=3601 or FINANCE on the TYPE macro:
Describes the components that compose this workstation. A maximum of four terminal cans can be defined. Acceptable values for components are:

36DS or FIDS

3604 display or 4704 keyboard/display

36DS3 or FIDS3

3604 display, model 3

36DS4 or FIDS4

3604 display, model 4

36DS7 or FIDS7

3604 display, model 7

36JP or FIJP

3610/3612 journal printer or 4710 receipt validation printer

36PB or FIPB

3611/3612 passbook printer

36FP or FIFP

3618 line printer

36MS or FIMS

Magnetic strip encoder

36CT or FICT

3614 customer transaction facility (a component of FINANCE)

One 36DS, 36DS3, 36DS4, 36DS7, or 36CT (or equivalent components specified as FIDS, FIDS3, FIDS4, FIDS7, or FICT) can be specified per workstation. Multiple components of the same type (excluding 36DS, 36DS3, 36DS4, and 36DS7 or equivalent F_{lxx} components) can be specified. The order in which components are specified determines the value placed in the COMPT field of the Finance Communication System output message header.

COMPT1=, COMPT2=, COMPT3=, COMPT4=

- For SLU 1 Terminals:
Defines the component types and processing associated with terminals defined as SLUTYPE1. A maximum of four components can be defined. For each component, three subparameters—media, edit processing, and features—can be defined.
You can assign a number to a set of user-defined features (such as printer size, vertical forms control, horizontal tabs on a printer, and logical record

TERMINAL Macro

length on a diskette) and tell MFS what these features are. IGNORE can be used to specify that all features (or any designated feature) are to be ignored.

Component 1 must be either CONSOLE or PRINTER1. Components 2, 3, and 4 must be PRINTER2, PRINTDS1, TRANSDS1, READER1, PUNCH1, or USERDS1.

If no components are specified, the default is COMPT1=(CONSOLE,BASIC-SCS1).

The first subparameter (media) of the second, third, or fourth components has no default.

If BASIC-SCS1, BASIC-SCS2, BASIC, or BASIC-TRN is specified as the second subparameter (editing), the third subparameter (feature) cannot be specified. If MFS-SCS1 or MFS-SCS2 editing is specified as the second subparameter, the default is IGNORE for the third subparameter.

The following values can be defined:

CONSOLE (BASIC-SCS1 or MFS-SCS1)

Identifies the presence of a console keyboard and a printer. Either this component or PRINTER1 must be (and can only be) specified first in a sublist, because all error messages are routed to this destination. This component uses SCS1 data streams. This can be the only component specified if the secondary logical unit type 1 is a 3767.

CONSOLE (BASIC-TRN)

Identifies a SLU 1 terminal whose input data does not conform to SCS (SCS1 or SCS2) protocols. IMS processes the data without editing, translating, or deleting characters following the destination and password fields. No MFS editing is supported.

This form of transparency support differs from SCS2 transparency support in that BINDPSB1=BINTRNDS in the SLU 1 bind image need not be set (and has no effect if set), and in that IMS does not scan the input data stream for transparency control characters (X'35') to identify transparent fields.

PRINTDS1

Identifies the presence of a print data set or the SYS.INTR data set for the SLU 1 programmable model. This component uses SCS1 data streams.

PRINTER1

Identifies the presence of 3784, 3284, 3286, or similar printers. This component or console must be (and can only be) specified first in a sublist, because all error messages are routed to these destinations. This component uses SCS1 data streams.

PRINTER2

Identifies the presence of a SLU 1 printer. This component uses SCS1 data streams.

PUNCH1

Identifies the presence of a 3770 card output device. This component uses SCS2 data streams. Card records do not span RU boundaries.

READER1

Identifies the presence of a 3770 card input device. This component uses SCS2 data streams.

TRANSDS1

Identifies the presence of an input component consisting of the SLUTYPE1 transmit data set. This component uses SCS2 data streams. TRANSDS1 can be specified only once as a component of a particular terminal.

USERDS1

Identifies the presence of the user's sequential data sets for output and input. MFS support is not available for this component. The data stream on output is undefined; the inbound data stream is the same as the transmit data set data stream.

If multiple USERDS1 components are defined on the same TERMINAL macro statement, all user data set (UDS) input appears to be from the first UDS defined. Also, unless the CHNG call is used by the IMS Message Processing Program, all output is sent to the output component associated with the first UDS defined. (See the COMPT parameter of the NAME macro statement: 131.)

If MFS-SCS1 or MFS-SCS2 data streams are specified, MFS-SCS1 or MFS-SCS2 formatting is used for messages processed by MFS.

- For SLU P nodes:

Defines the component types and processing (including distributed presentation management for MFS) associated with a node defined as SLUTYPEP, and indicates that the component is represented by a user-written program.

Up to four components can be specified; for each component, up to three subparameters can be specified. If the component is not specified, the default is COMPT1=(PROGRAM1,BASIC). Components 1, 2, 3, and 4 must be either PROGRAM1 or PROGRAM2. If component 2, 3, or 4 is present and takes all defaults (PROGRAM1,BASIC), at least one of the default subparameters must be specified so that the presence of the component can be recognized.

The first subparameter is the media type coded as PROGRAM1 or PROGRAM2.

If PROGRAM1 is specified, IMS does not assume component protection for this component and might send consecutive messages without waiting for intervening input requests. When using MFS, component protection can be accomplished for messages which use MFS by means of paging options in MFS control blocks defined as DEV TYPE=DPM-A01...DPM-A15 or DPM-A1...DPM-A15.

If PROGRAM2 is specified, IMS assumes component protection and does not send consecutive messages without intervening input requests.

The second subparameter specifies the type of editing to be provided by IMS for input to and output from user-written programs.

If BASIC is specified, no deblocking occurs on input, and MFS is not used on output or input.

If DPM-A01...DPM-A15 or DPM-A1...DPM-A15 is specified, no deblocking occurs on input, and the MFS DPM function can be used on output or input. To define DPM for this component, code DPM-An, where "n" is a decimal number from 1 to 15. When MFS is to be used, this operand must match an MFS DEV TYPE=DPM-An statement within the MFS definitions.

If BASIC-SCS1 is specified, SCS1 data streams are used. MFS is not available on output or input.

TERMINAL Macro

If MFS-SCS1 is specified, SCS1 data streams are used, and MFS is available.

The third subparameter specifies a user-defined feature code. The designated feature is used to select an MFS format—MFS-SCS1, DPM-A01...DPM-A15, or DPM-A1...DPM-A15—with matching feature specification. IGNORE can be used to specify that an MFS format (SCS1 or DPM-An) with the FEAT=IGNORE operand of the DEV statement is to be selected.

Related Reading: See *IMS/ESA Utilities Reference: Transaction Manager* for additional information on the MFS format.

The third subparameter cannot be specified if BASIC or BASIC-SCS1 is specified for the type of editing (second subparameter).

- For LU 6.1 nodes:

Defines the SNA send/receive and bracket protocols to be used during asynchronous IMS output to LU 6.1 nodes. The MFS distributed presentation management (DPM) facility can be defined as available selectively by component.

The first, second, and third subparameter are specified independently of one another.

The first subparameter (MULT1, MULT2, SINGLE1, SINGLE2) indicates the output protocol to be used and the number of messages to be sent on output. These subparameters are defined as follows:

- MULT1—Multiple messages or empty queue ends the bracket
- MULT2—Multiple messages or empty queue causes the flow to be returned to the other half-session
- SINGLE1—A single message or empty queue ends the bracket
- SINGLE2—A single message or empty queue causes the flow to be returned to the other half-session

The default is MULT1.

Related Reading: Guidance on specification of a component as SINGLE1, SINGLE2, MULT1, or MULT2 is found in the *IMS/ESA Administration Guide: Transaction Manager*, and the characteristics of SINGLE1, SINGLE2, MULT1, MULT2 are described in *IMS/ESA Customization Guide*.

The second subparameter (VLVB/DPM-Bn) indicates whether MFS is available for the component and whether the data blocking algorithm is to be used. VLVB indicates variable-length, variable-blocked format is to be used in place of MFS for both input and output. DPM-Bn indicates that MFS DPM is available for both input and output from that component on a message-by-message basis. The default is VLVB. You should not specify the DPM-B(xx) operand for the XRF/ISC link. If DPM-B(xx) is specified, a G732 warning message is issued, and the default value of VLVB is assumed.

The third subparameter, a number from 1-10, specifies a user-defined feature code. The designated feature is used to select an MFS format. IGNORE can be used to specify that an MFS format with the FEAT=IGNORE operand of the DEV statement is to be selected. The default is IGNORE. (If the second subparameter is specified as VLVB, the third subparameter cannot be specified.)

Related Reading: For more information, refer to *IMS/ESA Utilities Reference: Transaction Manager*.

EDIT=

The first parameter specifies whether (YES) or not (NO) the user-supplied physical terminal output edit routine specified in the TYPE or LINEGRP macro statement is to be used for this workstation. The default is NO.

The second parameter specifies whether (YES) or not (NO) the user-supplied physical terminal input edit routine specified in the TYPE or LINEGRP macro statement is to be used for this workstation. The default is NO.

FEAT=

(If the system definition statements from previous releases of IMS are used, TERMINAL macro statements from previous releases are valid for this release.)

- For 2741 terminals:

With the parameter INTERRUPT or NOINTRPT (the default), specifies whether the 2741 terminal is equipped with the Receive Interrupt feature.

- For 2780 terminals:

With the indicated parameters, specifies 2780 terminal features.

The first subparameter, (MULTIPLE or M) or (STANDARD or S), specifies the MULTIPLE or STANDARD record transmission feature. The default is STANDARD.

The second subparameter specifies the size of the print line. The default is 80.

The third subparameter specifies the terminal model number. The default is 1.

- For nonswitched or local 3270 terminals:

The first subparameter specifies whether a DEKYBD (Data Entry Keyboard) or PFK (Program Function Keys) is available on this 3275 or 3277 terminal. PFK and DEKYBD are not valid, and NOPFK (no Program Function Keys) is forced for 3284 and 3286 terminals. The default is PFK.

The second subparameter specifies whether the Operator Identification Card Reader is available (CARD) or not (NOCD) on this 3275 or 3277 terminal. CARD specification is not valid, and NOCD is forced for 3284 and 3286 terminals. The default is CARD.

The third subparameter specifies whether (PEN) or not (NOPEN) the Selector Pen is available on this 3275 or 3277 terminal. PEN specification is not valid and NOPEN is forced for 3284 and 3286 terminals. The default is PEN.

- For terminals defined as SLUTYPE2:

The comments for nonswitched and local 3270 terminals apply equally to terminals defined as SLUTYPE2.

- For all 3270 Terminals:

IGNORE can be specified if the presence or absence of the value does not affect terminal operation within IMS.

If IGNORE is not specified, the default values assumed for any omitted subparameters are PFK, CARD, and PEN, respectively.

The CONFIG macro is used to define these features for switched 3275s.

- For all 3270 terminals (with the exception of the printer component of the 3275) and secondary logical units type 2 (SLU 2):

TERMINAL Macro

F1, F2...F10 specify terminal features that the user designates to correspond to each integer. The same number is used for any given feature during both system definition and MFS processing. This specified integer is used instead of, and not in addition to, other feature specifications.

If you are using the same MFS format for the 3180 and 3290, Fn must have different values for the 3180 and 3290. These same values must be coded on the corresponding FEAT=Fn parameter of the MFS DEV statements.

- For 3275 printer components:

FEAT=F1...F10 is not a valid specification for 3275 printer components (that is, devices specified as UNIT=3275 and COMPT=PTR1).

- For 3284 and 3286 units:

F1...F10 specify terminal features that the user designates to correspond to each integer. The same number is used for any given feature during both system definition and MFS processing. This specified integer is used instead of, and not in addition to, other feature specifications.

Specifications of FEAT=F1...F10 and PTRSIZE are mutually exclusive for the 3284 and 3286 devices. If PTRSIZE= is specified, FEAT= is invalid. If FEAT= is specified, no default PTRSIZE is assumed.

Also, FEAT=IGNORE is invalid for these devices. PTRSIZE=IGNORE is used instead.

The use of a particular specification for F1...F10 should be directly related to a particular printer line size and should be consistently used for printers having that line size and identical attributes throughout your system definition.

Example: FEAT=F5 might be used to define printers with a line size of 126 print positions and color, FEAT=F6 for printers with a line size of 126 print positions but without color, and FEAT=F7 for printers with a line size of 120 or 132.

- For UNITYPE=3601 or FINANCE and COMPT=36FP:

DUAL indicates the presence of the DUAL forms control feature, and 132 indicates the presence of the extended print line feature.

- For SPOOL Terminals:

If FEAT=AUTOSCH, the print utility is automatically scheduled when the spool lines are stopped, when a spool data set fills, or when IMS is initialized. If FEAT=NOAUTOSCH (the default), the MTO must schedule the print utility.

- For System/3 and System/7:

When FEAT=INPUT, the terminal is treated as an input-only terminal and cannot receive output.

FPBUF=

Specifies the Fast Path terminal buffer size for the following terminal types: 2740 station control, 2740 Model 2, 3270 (BTAM and VTAM), 3601, SLU 1, SLU 2, NTO, SLU P, and LU 6.1. FPBUF provides the system default for the EMH buffer if the EMHL parameter is not specified. FPBUF=0 means that the terminal is not eligible for Fast Path processing. FPBUF=0 is the default. This specification can also be used to override an FPBUF= default on the TYPE macro statement. If the terminal is eligible for Fast Path transaction processing, the minimum value that can be specified is 12. The maximum value is 30632. The value specified must not be greater than the logical record length of the long message queue data set specified in the RECLNG= parameter of the MSGQUEUE macro.

TERMINAL Macro

This keyword is optional and cannot be specified on ETO terminal descriptors. See the RECLNG parameter of the MSGQUEUE macro for further information.

HEADCTL=

Specifies that the header control feature for this 2740 Model 2 terminal is valid only for 2740 Model 2 lines, and specifies the number of characters reserved for header control. If either the BUFSIZE or HEADCTL operand is omitted, the default value is the value specified on the preceding LINE statement. An entered value for header control cannot exceed 28 and must be a multiple of 4.

LTERM=

Specifies a logical terminal name for this TERMINAL statement. This name is used to check input terminal security for this terminal. The NAME macro referred to must precede this TERMINAL statement. When this operand is present, the TERMINAL statement cannot be followed by a NAME statement.

This operand is required for READER terminals. For example, a previously defined TERMINAL statement defined a 3270 remote terminal and was followed by the name ABLE. In addition, assume that this TERMINAL statement refers to a READER terminal that is an input-only device. All input security for the READER terminal is checked against the security characteristics assigned to ABLE.

If a READER terminal requires a dummy terminal definition for the NAME macro, code the dummy definition so that it is consistent with BTAM use in the system. The UNITYPE parameter of the LINEGRP macro determines whether BTAM is to be included.

This operand **cannot** be specified for NTO devices and is not supported for ACF/VTAM terminals.

When a list of LTERMs is defined under the previous TERMINAL statement to which reference is made, only the first LTERM name is selected from that list. If an attempt is made to select other than the first, IMS ignores the name chosen and assigns the value of the first name listed.

Example:

```
TERMINAL  ADDR=E2
  NAME    LTERM1
  NAME    LTERM2
  NAME    LTERM3

TERMINAL  ADDR=E4, LTERM=LTERM2
```

In this example, IMS treats the second TERMINAL statement as though it reads LTERM=LTERM1.

MODEL=

Specifies the terminal model number for terminals with screen sizes of 480 (Model 1) or 1920 (Model 2) characters. Terminals with screen sizes other than 480 or 1920 characters are defined using the TYPE and SIZE keyword parameters. The default is 1. The model number specified for a 3277 for which OPTIONS=COPY is specified must be less than or equal to the model number specified for any candidate printer. The MODEL and TYPE/SIZE keyword parameters are mutually exclusive.

TERMINAL Macro

MODETBL=

Specifies the name of the VTAM logon mode table entry (logon mode name) containing the SNA bind parameters to be used when a session is established for this terminal.

This function allows a system definition specification for referencing an entry other than the default entry in the user's VTAM logon mode table. Normally, the terminal operator specifies this mode table entry name when logging on to a terminal; this is not possible, however, from the IMS master terminal because IMS initiates the session.

With this function, if MODETBL= is not specified at system definition, no functional or operational change affects the user.

If MODETBL= is specified at system definition, the specified entry name is used. The MODETBL can may be overridden by:

- The LOGON APPLID entry by the remote terminal operator
- VARY ACT, LOGON= command by the network terminal operator
- /OPNDST, /RST, or /CHANGE command by the MTO

The MODETBL= parameter is required for the IMS master terminal if the VTAM default mode table is not configured specifically for the device to be used as the IMS master terminal.

Related Reading: For device dependent considerations, see *IMS/ESA Administration Guide: Transaction Manager*.

With MODETBL=, you do not need to specify the mode table entry when logging on to terminals that always require specification of the same mode table entry name.

MSGDEL=

Specifies which message types IMS should discard for this terminal, if it is not the master terminal. The default is NONE for non-VTAM terminals and SYSINFO for VTAM terminals. NONE is ignored for VTAM. The MSGDEL option applies to a switched terminal only while that terminal is connected to IMS; any message queued while an LTERM is not connected to the switched line is not deleted.

NONIOPCB | NOTERM | SYSINFO | NONE

NONIOPCB specifies that IMS should discard the following message types destined for this terminal:

- Message switches
- Messages inserted by an application program to an alternate PCB
- /BROADCAST messages
- DFS059 TERMINAL status messages

When a switched terminal defined as MSGDEL=NONIOPCB is not connected to IMS, any message normally discarded as a function of specifying NONIOPCB is queued for an LTERM that might become associated with the switched terminal at signon (/IAM) time.

NOTERM specifies that IMS should discard DFS059 TERMINAL status messages and DFS3650 SESSION status messages for this terminal. Do

not specify NOTERM for 3601, SLU P, or LU 6.1 VTAM terminals. If NOTERM is specified, a warning message is issued and the specification default is SYSINFO.

SYSINFO specifies that IMS should discard DFS059 TERMINAL status messages for this terminal.

NONE specifies that IMS should not discard any message types for this terminal.

When MSGDEL is specified, the following additional restrictions apply to the reassignment of logical terminals:

- Assignment of a logical terminal to a physical terminal defined as MSGDEL=NONIOPCB can only take place if no messages are in the Q3 (logical terminal's system) or Q4 (other than system) queues, unless that logical terminal is currently assigned to a physical terminal also defined as MSGDEL=NONIOPCB.
- Assignment of a logical terminal to a physical terminal defined as MSGDEL=SYSINFO or MSGDEL=NOTERM can only take place if no messages are in the system queue of the logical terminal, unless that logical terminal is currently assigned to a physical terminal defined as MSGDEL=SYSINFO, MSGDEL=NOTERM, or MSGDEL=NONIOPCB.

An LU 6.1 session can be initiated only if the MSGDEL specifications on both the TERMINAL macro statement and the SUBPOOL macro statement match. An LU 6.1 session is not established, and an error message is issued if these specifications are not identical.

NAME=

Must be the VTAM node name specified during VTAM/NCP generation. To avoid a possible conflict in using Security Maintenance Utility (SMU) STERM statements, do not use the names LINE and ALL as a VTAM node name .

When using XRF and VTAM-supported master terminals, a second node name should be specified for the master terminals. The two node names specify nodes in the two IMS XRF systems. The first is used by the system with HSBID=1 and the second by the system with HSBID=2. If a secondary master is used, it also should have two specifications. For more information on the HSBID parameter, see "Chapter 4. Procedures" on page 189.

When using RSR and VTAM-supported master terminals, a third node name can be specified for the master terminals. The three node names specify nodes in the IMS RSR subsystems. The first is used by the subsystem with MTOID=1, the second by the subsystem with MTOID=2, and the third by the subsystem with MTOID=3. If a secondary master is used, it also should have at least two specifications (for RSR without XRF) or three specifications (for RSR with XRF at the active site). Note that the third node name can only be used for the tracking subsystem. For more information see the description of the MTOID parameter for "Member DFSSRxx" on page 321.

The specification of *nodenamen* corresponds to the HSBID= specification at IMS startup. This means that HSBID=2 uses *nodename2* as the master or secondary master terminal, and uses PASSWD2 (from COMM macro). RSR tracking subsystems must select *nodenamen* and PASSWDn using the MTOID= parameter in the DFSSRxx member. Tracking subsystems can override the APPLID for the subsystem by using the APPLID3= startup parameter.

TERMINAL Macro

When you define an ISC surveillance link between the XRF active and alternate systems, the two node names in the NAME keyword must match the two VTAM application names in the APPLID keyword of the COMM macro.

For an ISC node communicating with an XRF complex, the node name must be the VTAM USERVAR associated with the XRF complex. Defining additional nodes for the actual VTAM APPLIDs of the IMS systems which comprise the XRF complex will result in errors during session initiation.

The recommended use of the *nodenamen* keyword is:

- Use *nodename1* only for DB/TM or DCCTL systems that are non-XRF and non-RSR
- Use *nodename1* and *nodename2* for DB/TM or DCCTL systems using XRF
- Use *nodename1* and *nodename3* for DB/TM or DCCTL systems using RSR
- Use *nodename1*, *nodename2*, and *nodename3* for DB/TM or DCCTL systems using both XRF and RSR

The default for *nodename3* is the value of *nodename1*.

OPTIONS=

Specifies certain communication options associated with this terminal. The OPTIONS= subparameters are not position dependent.

ACK | **OPTACK**

ACK indicates that recoverable and update transactions must be acknowledged with a definite response 1 (FME) or definite response 2 (RRN). OPTACK allows input messages containing only a begin bracket indicator to be acknowledged by an end bracket indicator on the next output message. A definite response must be requested for recoverable input if ACK is specified, and is optional if OPTACK is specified. The use of OPTACK also affects the sequence number recording for message resynchronization.

Related Reading: Additional information on ACK and OPTACK is found in *IMS/ESA Administration Guide: Transaction Manager*.

OPTACK is forced for Fast Path-eligible terminals if the value specified for FPBUF is greater than 0. ACK can be specified if BID is not also used. ACK allows a 3600/LUP terminal to add Fast Path without initially recoding the remote terminal program.

BSELM | **NBSELM**

Specifies whether (BSELM) or not (NBSELM) the backspace character elimination is required for input from a SLU 1 terminal.

COPY | **NOCOPY**

Specifies whether (COPY) or not (NOCOPY) the copy function is requested for this terminal. Terminals defined with the copy function must have appropriate candidate printers.

Related Reading: For a detailed discussion of candidate printers for 3270 terminals, refer to the discussion of controlling output in *IMS/ESA Administration Guide: Transaction Manager*.

This subparameter is supported for dynamic SLU 2 terminals, but not for dynamic non-SNA VTAM 3270s.

FPACK | NFPACK

Applies only to 3601/Finance and SLU P terminal types and specifies whether (FPACK) or not (NFPACK) Fast Path output message protocol is to be used, as defined in *IMS/ESA Administration Guide: Transaction Manager*.

The Fast Path output message protocol for FPACK normally requires another input (transaction or SNA RTR command), following transmission of the output Fast Path reply message, before any asynchronous output message can be sent. The NFPACK option indicates that standard non-Fast Path protocols are to be used, and allows any asynchronous output message to be sent immediately following acknowledgment of the Fast Path output reply message.

NOASR | ASR

For the Session Outage Notification facility only; specifies automatic session restart (ASR) on a VTAM node. The default is NOASR. Specifying ASR or NOASR on the TERMINAL macro overrides ASR definitions on the TYPE macro for a defined node.

Use the /DIS command to display the current ASR option, and the /CHANGE command to change it.

The ASR option is not supported for nodes that can have XRF backup sessions.

For LUTYPE6 nodes, the ASR option works only if IMS is acting as the primary logical unit (PLU).

NOBID | BID

NOBID indicates that the VTAM BID command is never to be used. BID indicates that the VTAM BID command always precedes output messages that occur while between brackets, regardless of response mode option used.

Prior to the availability of this option, IMS systems forced the equivalent of NOBID if NORESP was specified, and forced the equivalent of BID if FORCRESP is specified.

NODISCON | DISCON

Determines whether IMS should automatically terminate a session for a terminal defined as SLUTYPE1 when operating in unattended mode. (ATTEND or UNATTEND mode is determined from the logon mode table entry specified to be used during session initiation.) DISCON specifies that IMS should automatically terminate the communication session with the terminal after receiving all immediately available input or sending all available output. NODISCON suppresses that automatic termination. The logon mode table is described in the MODETBL= keyword description.

NOFES | FES

Specifies whether (FES) or not (NOFES) this terminal is capable of front-end switching. The default is NOFES.

To define a terminal as capable of front-end switching, define the front end switch exit routine with the FESEXIT keyword of the COMM macro. See "COMM Macro" on page 64 for more information about the FESEXIT keyword.

TERMINAL Macro

If FES is specified, the FES exit will get control for each input from this terminal.

NOMFS | **MFS**

Specifies whether (MFS) or not (NOMFS) the message format service is to be provided for this terminal. The default is NOMFS.

NOMTOMSG | **MTOMSG**

Specifies whether (MTOMSG) or not (NOMTOMSG) normal session initiation and termination messages are to be sent to the IMS MTO. The default is NOMTOMSG.

NOSHARE | **SHARE**

Specifies whether (SHARE) or not (NOSHARE) this printer can be shared between VTAM subsystems. If SHARE is specified, then IMS can issue a SIMLOGON VTAM macro when either of the following occurs:

- Data is queued for this printer.
- The VTAM 3270 PF12 key invokes the internal IMS copy function that selects this terminal, and the terminal is not connected to IMS.

If SHARE is specified, the OPNDST option must also have been selected for this terminal.

If SHARE is specified, a MODETBL entry cannot be specified for cross-domain resources. An entry of ALL X'00' in the VTAM MODETBL indicates to ACF/VTAM that the default session parameters are to be used for this terminal and that the terminal can be defined in any VTAM domain.

If the default option, NOSHARE, is specified, and if IMS releases this terminal to another VTAM application, the IMS MTO must issue an /OPNcommand, a VTAM VARY command, or a USS logon in order for IMS to reacquire the terminal.

This option is applicable to SLU 1, 3284/86 local, and VTAM 3270 devices only. If UNITYPE=3270 is specified on the TYPE macro statement, UNIT must be a 3284 or 3286 device.

NOXPAR | **XPAR**

Specifies whether (XPAR) or not (NOXPAR) transparent data transmission is to be used for this 3270 terminal. Specification of XPAR indicates that MFS formatting is to be bypassed for this terminal. This option applies only to BTAM remote 3270 terminals. 3290 terminals **must** specify XPAR.

OPNDST | **NOPNDST**

Applies only to VTAM terminals and specifies whether (OPNDST) or not (NOPNDST) the /OPNDST command is valid for this node. The default for all terminal types except NTO is that /OPNDST is valid.

The default value for NTO devices is NOPNDST. Either NOPNDST, the default value, or OPNDST is valid for nonswitched NTO devices. When using switched NTO devices, NOPNDST must be specified or accepted by default, because these devices must be logged on by an operator. An error results if OPNDST is specified for switched NTO devices.

The NOPNDST option is used to prevent the /OPNDST command from being used to start a session between IMS and a VTAM terminal.

This option does not apply to a VTAM terminal specified in the NAME macro as MASTER (the IMS master terminal), because that terminal is automatically opened.

PAGDEL | NPGDEL

Specifies whether (PAGDEL) or not (NPGDEL) automatic page deletion is to be specified for this terminal. This option applies to 3270 terminals, SLU 2 terminals, SLU P nodes, and to 3601 workstations using MFS. PAGDEL is the default option, but the LU 6.1 nodes default is NPGDEL.

When the last physical page of the last logical page of the message is sent, the message is dequeued no matter which parameter is specified.

PAGDEL should be specified for terminals used in an interactive or conversational manner. If input from the terminal or node is normally followed by a response to the terminal or node, PAGDEL should be specified.

NPGDEL is provided for terminals that produce many input messages to which no response is made. When NPGDEL is specified, input from a paged message is responded to by IMS with the first page of that same message. Any application program response is queued.

PROT | UNPROT

Specifies whether (PROT) or not (UNPROT) the terminal display is protected after each page of a message. If PROT is specified, the protected status of the terminal depends upon the unprotected screen mode option on the DSCA keyword of the MFS DEV statement or the SCA keyword of the MFS MFLD statement. This keyword applies only to 3270 display terminals and secondary logical units type 2 (SLU 2).

RELQR | NORELQR

Specifies whether (RELQR) or not (NORELQR) IMS is to release this terminal upon request from other VTAM subsystems. If NORELQR is specified, requests by other VTAM subsystems for this terminal are not honored. The MTO must intervene to reassign this terminal to another VTAM subsystem. If RELQR, the default, is specified, then IMS releases this terminal to other VTAM subsystems upon request.

This option is applicable to devices specified on the TYPE macro statement as UNITYPE=3601, FINANCE, SLUTYPE1, SLUTYPEP, and LUTYPE6, in addition to 3284 and 3286 printers.

NORELQR is forced for 3277, 3275, SLU2, and NTO terminals.

SCAN | NOSCAN

Specifies whether (SCAN) or not (NOSCAN) IMS should edit 3601 device-control sequences (such as SELECT, POSITION, and WRITE TRANSPARENT) that would otherwise be split across two transmissions when a non-MFS formatted output segment must be split into multiple transmissions.

If an MFS-formatted output segment must be split across two transmissions, IMS edits 3601 device-control sequences (SELECT and POSITION) that would otherwise be split across two transmissions even if the NOSCAN option is specified.

TERMINAL Macro

Related Reading: See “Financial Communication System Programmer’s Guide” in *IMS/ESA Administration Guide: Transaction Manager* for additional information.

SYNCSSESS | FORCSSESS

Specifies when session initiation is to be completed.

- SYNCSSESS specifies that session initiation is to be completed only when session restart modes (message sequence numbers) agree.
- FORCSSESS specifies that session initiation is to be completed regardless of the agreement of session restart modes.

The default is SYNCSSESS. only when session restart modes (message sequence numbers) agree (SYNCSSESS) or whether initiation should always be completed regardless of the agreement of session restart modes (FORCSSESS). The default is SYNCSSESS.

TRANRESP | NORESP | FORCRESP

TRANRESP specifies that the transaction code definition should be referred to determine whether response mode should be used. All Fast Path-eligible terminals operate in response mode—either TRANRESP or FORCRESP.

- TRANRESP is the default value for 2740-1, 3767, 3770 switched and station-control terminals, and for terminals defined as SLUTYPE1, NTO, or LUTYPE6.
- NORESP specifies that response mode is not to be allowed for this terminal. NORESP is the default for 2740-2, 3270, and 3601 terminals, and terminals defined as SLUTYPE2 and SLUTYPEP. NORESP is not compatible with Fast Path.
- FORCRESP specifies that response mode is to be forced for all transactions entered from this terminal. FORCRESP is the default for 2740 non-station control and 2741 terminals.

For BTAM and VTAM terminals, the combined specification of FORCRESP and NPGDEL is not recommended. Also, if MSGTYPE=RESPONSE is specified on the TRANSACT macro statement for these terminals, the combined specification of TRANRESP and NPGDEL is not recommended.

The keyboard of a display terminal defined with no automatic page deletion (NPGDEL) can become locked during execution if operating in terminal response mode. NPGDEL specifies that the current output message is not to be dequeued at the time of input. However, with terminal response mode, input is inhibited until the current output message is dequeued and the terminal response mode is reset. Thus, a terminal receiving multiple pages of output and defined with the above combination of specifications might require master terminal intervention to reset terminal response mode.

TRSOSI | NOTRSOSI

Specifies whether IMS translates the shift-out/shift-in (SO/SI) characters for MFS editing. The SO/SI characters indicate which characters are used for EGCS terminals. The default of TRSOSI should be used for non-EGCS terminals. This keyword only applies to 3270 terminals and type 2 secondary logical units (SLU2).

UNLOCK | LOCK

Specifies whether IMS is to unlock the terminal keyboard and reset the modified data tags after an MFS bypass.

- UNLOCK specifies that MODNAME=DFS.EDTN unlocks the terminal keyboard and resets the modified data tags
- LOCK specifies that an application program assumes responsibility for unlocking the terminal keyboard and resetting the modified data tags.

This keyword applies only to 3270 terminals and type 2 secondary logical units (SLU2).

OUTBUF=

Specifies the size of the IMS output buffer to be used for the work stations described below. OUTBUF indicates the maximum size of IMS message segments that are sent from IMS to the adapter in a single transmission, except for BTAM devices operating in BSC mode and specified with data transparency (OPTIONS=XPAR).

Because the adapter converts IMS message segments into APPC logical records, the OUTBUF size is also the maximum size of APPC logical records sent by the adapter to the APPC application. If an IMS message segment is larger than OUTBUF, IMS breaks the message into pieces of size OUTBUF. When this occurs, the APPC application must issue multiple MC_RECEIVE_AND_WAIT verbs to receive the message (because each MC_RECEIVE_AND_WAIT receives a single logical record).

For VTAM devices, a single transmission represents a transmission from IMS to VTAM. For these devices, the output message is blocked based on the OUTBUF value.

For remote 3270 display devices specified with data transparency (OPTIONS=XPAR), the output message is blocked based on the OUTBUF value. For VTAM non-SNA devices, BTAM local 3270 devices, and remote 3270 devices specified without data transparency (OPTIONS=NOXPAR), the OUTBUF value must be large enough to contain the entire output message. When sending to a printer, OUTBUF must be equal to or greater than the segment size, but less than 4096 bytes.

IMS also uses the OUTBUF value to set the transmission services usage field (byte 11) in the bind parameter. If this value is specified incorrectly, it might cause the device to reject the bind parameter. The value specified, plus the size of any routing information later appended by VTAM, must be less than the maximum message size specified for VTAM and its related NCPs. The output buffer size is calculated as follows for:

3270: Record size. The range of values that can be specified is 256 through 30720 bytes. The default is 2000 bytes. The actual hardware buffer capacity should be considered when creating the OUTBUF calculation. The hardware buffer maximum might be less than the default of 2000 bytes. For a device that is not a SNA 3270, OUTBUF is used only if you are using MFS bypass. Otherwise, OUTBUF is ignored.

Related Reading: For information on devices attached to a local 3274 control unit, see *3274 Control Unit Description and Programmer's Guide*.

3601 or FINANCE: Record size plus header. (Header is 3 bytes if MFS is not used or 29 bytes if MFS is used.) The range of values that can be specified is 64 to 30720 bytes. The default is 64 bytes.

TERMINAL Macro

SLU 1: Record size plus header. (The header is 25 bytes.) The range of values that can be specified is 128 to 30720 bytes. The default is 256 bytes.

NTO: The range of values that can be specified is 256 to 30720 bytes. The default is 256 bytes.

Output message segments are not blocked. Each segment is sent as an RU. Segments greater than the RU size are spanned, but an RU never contains more than one segment.

SLU 2: The range of values that can be specified is 256 to 30720 bytes. The default is 1500 bytes. A 3790 defined as SLUTYPE2 has a maximum value of 1536.

SLU P: Record size plus header. (The header is 5 bytes if MFS is not used and up to 42 bytes if MFS is used.) The range of values that can be specified is 64 to 30720 bytes. The default is 64 bytes.

LU 6.1: Record size plus header. The range of values that can be specified is 256 to 30720 bytes. The default is 256 bytes. The header size without MFS is variable from 8 to 45 bytes; with MFS, the header size is variable from 8 to 62 bytes.

Related Reading: For more information on LU 6.1 record size, see *IMS/ESA Administration Guide: Transaction Manager*.

For all logical units that require a bind from IMS, the OUTBUF buffer size must be a decimal value that can be expressed by the algorithm $X \text{ times } 2 \text{ to the power of } Y$. X must be a value from 8 through 15 and Y must be a value from 3 through 11. This is a VTAM restriction. Thus, for example, the value 144 (representing 9×2^4) and the value 28672 (representing 14×2^{11}) are acceptable values. Details of this VTAM restriction are described in *ACF/VTAM Macro Language Reference*, with information on requesting unit sizes.

Valid VTAM buffer sizes are restricted to one of the following decimal values:

112	448	1792	7168	28672
120	480	1920	7680	30720
128	512	2048	8192	
144	576	2304	9216	
160	640	2560	10240	
176	704	2816	11264	
192	768	3072	12288	
208	832	3328	13312	
224	896	3584	14336	
240	960	3840	15360	
256	1024	4096	16384	
288	1152	4608	18432	
320	1280	5120	20480	
352	1408	5632	22528	
384	1536	6144	24576	
416	1664	6656	26624	

IMS converts the value given here to the format required for the bind parameter fields and places the value into the appropriate field in the bind parameter list. If the maximum RU value specified cannot be converted exactly into the bind

format, the value is rounded down to the next –lower value bind format for inbound RUs and rounded up to the next– higher value bind format for outbound RUs.

PTRSIZE=

Specifies the number of print positions of the 3284 or 3286 printer. The default is 120.

PTRSIZE=IGNORE causes MFS to always use the device output format (DOF) with FEAT=IGNORE in its DEV statement when editing output for this device. The existing parameters and control block values, including default parameters, are unchanged.

PU=

Specifies the type of physical device that is connected by means of the Network Terminal Option (NTO) licensed program. PU= is a required keyword for NTO devices; an error message results if it is omitted.

Each keyword represents a device type.

- TTY—TTY devices, such as a 33/35, or TTY compatible devices, such as a 3101 running in TTY mode.
- LUNS—These devices are non-SNA logical units, such as a 3101 not running in TTY mode.

SEGSIZE=

Specifies the maximum size of IMS message segments that are sent from the Adapter to IMS. Acceptable values are from 256 to 32000 bytes. The default is 256.

If an APPC logical record exceeds the SEGSIZE value, IMS will break the record into multiple IMS segments. The IMS application must issue get Next (GN) DL/I calls to receive the 2nd through nth segments.

Calculate the segment size as follows:

- SLU 1, SLU P, and NTO devices:
Size is the maximum input segment size that can result from blocking or deblocking one or more SCS1 input records.
- LU 6.1:
Size equals the size of the largest input record that can result from blocking or deblocking one or more VLVB or chain records.

SESSION=

For LU 6.1 sessions, specifies the maximum number of parallel sessions for the logical unit defined in the TERMINAL macro statement. From 1 to 4095 parallel sessions can be established. The default is 1.

When defining the XRF ISC SURVEILLANCE link, set the number of parallel sessions equal to one, or eliminate the SESSION= parameter.

SIZE=

Specifies the physical screen size of this 3270 display terminal having the device type symbolic name in the TYPE= operand described below. MFS uses the specified screen size to verify format specification. The first and second positional parameters of the operand relate to screen lines and columns, respectively. Both numbers must be specified as decimal numbers having the following range:

TERMINAL Macro

llll = 1 through 16384
cccc = 2 through 16384
product of llll x cccc = 80 through 16384

where *llll* is the number of lines and *cccc* the number of columns available on the screen.

For 3270 devices defined as SLUTYPE2, the maximum number of lines and the maximum number of columns that can be defined for each is 255. Any symbolic name can be associated with any size screen. However, after this association is defined, the same device type symbolic name must be used. That is, it must duplicate a previously specified TYPE= specification of the LINEGRP, TYPE, or TERMINAL statement.

Specification of the physical screen size is required for the first (or only) definition of the specific device type symbolic name and can be omitted for subsequent identical device type symbolic name specifications. If omitted, the SIZE keyword operand originally specified with the device type symbolic name is used when the symbolic name is repeated. If the SIZE keyword is specified, the screen size defined must be the same as that defined for the original device type symbolic name specification. You should establish a standard for relating device type symbolic names to the physical screen sizes.

The following standard is recommended:

User-Defined Symbolic Name	Screen Size
3270-A1 or 3270-A01	12x80
3270-A2 or 3270-A02	24x80
3270-A3 or 3270-A03	32x80
3270-A4 or 3270-A04	43x80
3270-A5 or 3270-A05	12x40
3270-A6 or 3270-A06	6x40
3270-A7 or 3270-A07	27x132
3270-A8 or 3270-A08	62x160

For the 3270 display devices defined with the device type symbolic name, the system definition produces an MFS Device Characteristics Table (DFSUDT0x) where x is the suffix specified in the SUFFIX keyword of the IMSGEN macro statement. You can also use the MFS DCT utility to build and update this table. This table is stored in IMS.RESLIB and is used by the MFS Language Utility and the IMS online utility to extract the information associated with the specific device type symbolic name.

Related Reading: Refer to *IMS/ESA Utilities Reference: Transaction Manager* for additional information on the MFS Language utility.

MNOTE statements in the system definition stage 1 output listing denote each device type symbolic name specified for 3270 and SLU type 2 devices with their corresponding screen size values.

Example:

Display Device Symbolic Names and Screen Sizes

Symbolic Name	Screen size	
	Lines	Columns
3270-A01	12	80
3270-A02	24	80
3270-A03	32	80
3270-A04	43	80

After a device type symbolic name (for example, 3270-A03) is associated with a screen size (for example, 32x80), the size need not be specified and must not be changed for any TYPE or TERMINAL macro statement for which the same device type symbolic name is used. During a subsequent system definition, the MFS Device Characteristics table and terminal control blocks can change if terminal specifications are altered. You should examine your MFS formats to determine whether they are affected by the changes and recompile them if necessary.

TYPE=

Specifies that an alphanumeric symbolic name for a device type is to be associated with this IMS-supported BTAM or VTAM 3270 physical display terminal. The symbolic name can be of the form 3270-*Ann*, where *nn* is a value from 01 through 15. The leading zero can be omitted. The physical screen size of the 3270 display device associated with the device type symbolic name is specified in the SIZE= operand, described above. Any 3270 display device can be defined with a device type symbolic name. 3270 display devices having 480- or 1920-character screen sizes can be defined using the MODEL keyword parameter rather than the TYPE and SIZE keyword parameters. However, 3270 display devices with other than 480 or 1920 characters must be defined using the device type symbolic name and, therefore, TYPE and SIZE. The TYPE/SIZE and MODEL keyword parameters are mutually exclusive.

UNIT=

Specifies the terminal for the previously defined line group. For 3270 line groups, UNIT= is used to define this terminal as a 3284 or 3286 printer or 3277 display unit. 3277 display units cannot be defined in the same local line group as 3284 or 3286 printers. This operand is also used to define a 3275 terminal that is part of a 3270 remote line group.

TRANSACT Macro

The TRANSACT macro statement is used one or more times with each APPLCTN macro statement to identify transactions as IMS exclusive, IMS Fast Path potential, or IMS Fast Path exclusive. It specifies the transaction codes that cause the application program named in the preceding APPLCTN macro to be scheduled for execution in an IMS message processing region. It also provides the IMS control program with information that influences the application program scheduling algorithm. It can define a message editing routine.

If this macro statement is preceded by an APPLCTN macro defining it as remote (specifies SYSID), this transaction is generated as a remote transaction.

An IMS Fast Path–exclusive transaction is identified by a TRANSACT macro statement following an APPLCTN statement that specifies both PGMTYPE=TP and FPATH=YES.

TRANSACT macro statements that identify IMS Fast Path–exclusive transactions generate routing code entries. These entries have the same value as the corresponding transaction codes. They are used by the IMS Fast Path Expedited

TRANSACT Macro

Message Handler routines to locate the Fast Path application program named in the preceding APPLCTN macro statement that processes the input message.

The SMU cannot define a Fast Path-exclusive transaction as able to issue commands.

An IMS Fast Path potential transaction is identified by a TRANSACT macro statement that specifies FPATH=YES following an APPLCTN statement that specifies PGMTYPE=TP and FPATH=NO.

An application defined as FPATH=NO cannot be run as a Fast Path region (IFP), even though it contains Fast Path potential transactions. Another application, defined with FPATH=YES, must exist and one of the following:

- A TRANSACT macro for the transaction
- A routing code defined by the RTCODE macro for the transaction

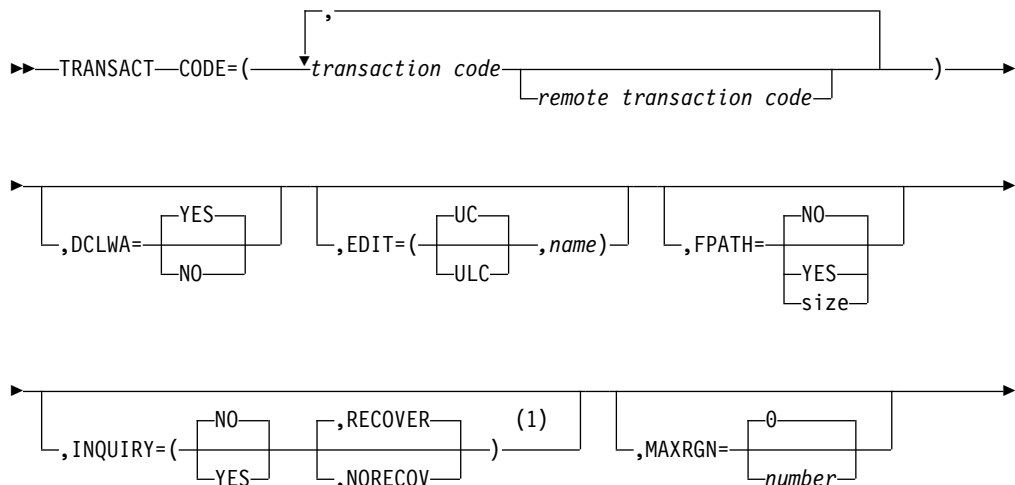
Related Reading: See the description of the Fast Path Routine for the Input Edit/Routing exit routine in the *IMS/ESA Customization Guide*.

All Fast Path transactions are implicitly defined as recoverable.

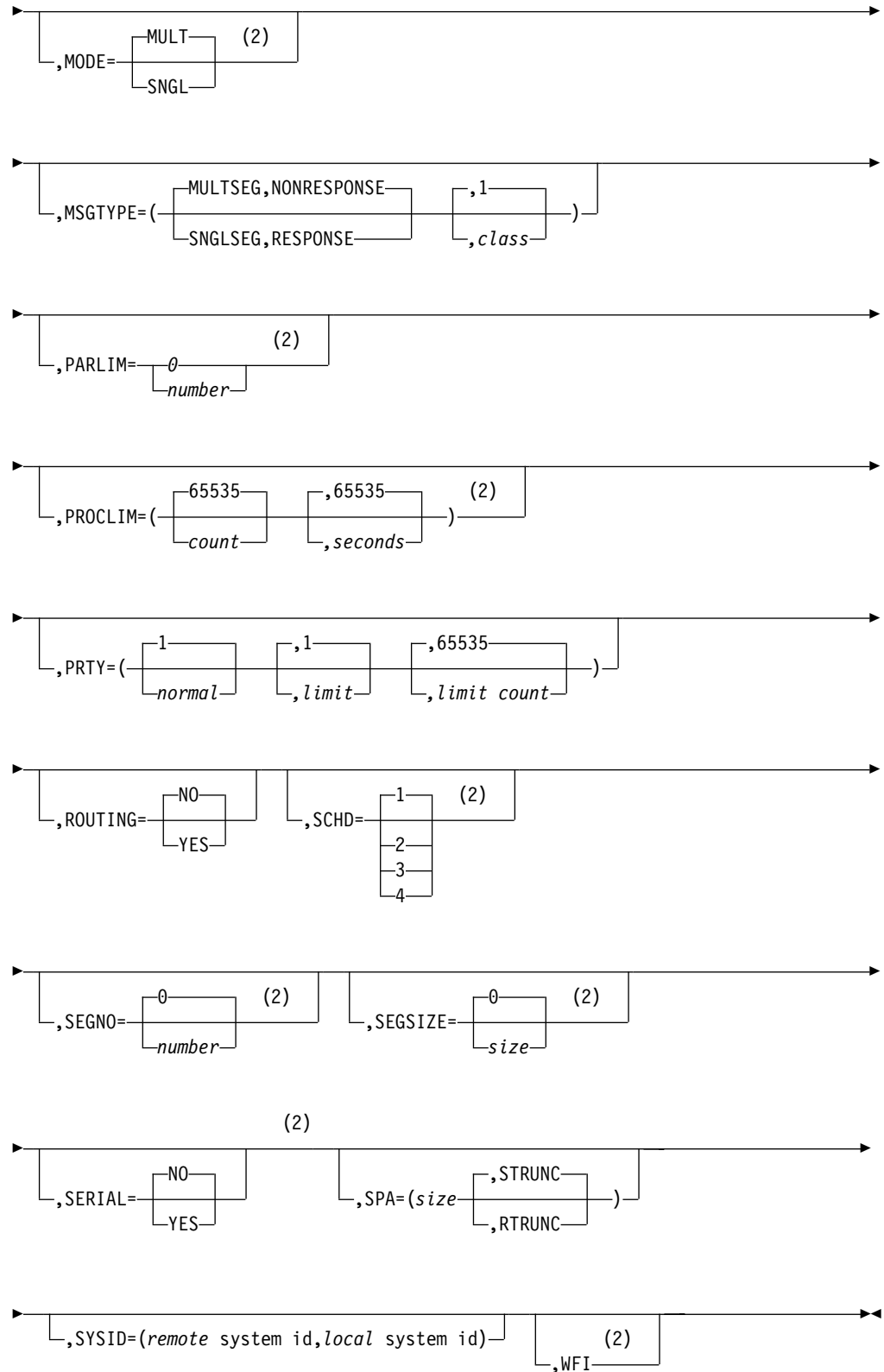
MSGTYPE=(SNGLSEG,RESPONSE) must be specified on the TRANSACT statement for Fast Path potential transactions.

Additional routing codes can be associated with a Fast Path application program by using the RTCODE macro statement following the corresponding APPLCTN macro statement.

TRANSACT macro statements with FPATH=YES and RTCODE macro statements are invalid after APPLCTN macro statements that define an IMS batch message processing (BMP) application.



TRANSACT Macro



Notes:

- 1 NORECOV is not valid when NO is specified.

TRANSACT Macro

- 2 These operands are not appropriate if a transaction destined for processing in another (remote) system is being specified.

Keyword Parameters

To find which parameters apply to your IMS configuration refer to Table 12 on page 32.

CODE=

Specifies one or more one- to eight-character alphanumeric transaction codes or remote transaction codes. If more than one code is specified, each transaction code name is assumed to have the same characteristics as all other keyword or positional parameter specifications or defaults. Enclose multiple names in parentheses and separate them with commas. Each character of transaction codes and logical terminal names must be an alphanumeric character (A through Z, #, \$, @, or 0 through 9). Transaction codes, logical terminals on the NAME macro, and linknames on MSNAME macros must comprise a set of values, each of which is unique in the system. That is, transaction codes, logical terminal names, and MSC linknames, collectively, cannot contain duplicates. The CODE operand is required.

Example:

```
CODE=(TRAN1,TRAN2,TRAN3)
```

DCLWA=

Specifies whether (YES) or not (NO) IMS should perform log write-ahead for recoverable, nonresponse mode input messages and transaction output messages. If not specified in the TRANSACT macro, the default is the DCLWA parameter in the IMSCTRL macro. The DCLWA parameter in the TRANSACT macro overrides the IMSCTRL macro parameter for this transaction.

Specify or accept a default of YES to ensure that both of the following occur:

- A nonresponse input transaction is made recoverable across IMS failures, prior to IMS acknowledging receipt of the input.
- Database changes are made recoverable prior to IMS sending associated output reply messages.

YES ensures that information in the log buffers is written to the IMS log, before the associated input acknowledgement or output reply is sent to the terminal.

Specify or accept a default of YES for all VTAM terminal types. For BTAM terminals, this option might increase input and output message path lengths.

Related Reading: For a information about the performance implications of selecting log write-ahead for transactions, see *IMS/ESA Administration Guide: System*.

Specify NO if input message integrity and the consistency of output messages with associated database updates is not required. DCLWA does not apply to response mode or Fast Path input processing, and is ignored during IMS execution.

EDIT=

Specifies whether (UC) or not (ULC) the input data is to be translated to uppercase. The first parameter of this operand defines whether the transaction

TRANSACT Macro

is uppercase/lowercase (ULC) as entered from the terminal, or if it is to be translated to uppercase (UC) before being presented to the processing program. The default is UC.

Specifying UC for VTAM terminals prevents the transmission of embedded device control characters.

You can also use EDIT to specify the one- to eight-character name of your own transaction input edit routine that edits messages prior to the program receiving the message. This name must begin with an alphabetic character. The specified edit routine (load module) must reside on the USERLIB data set prior to IMS system definition stage 2 execution. This routine cannot be the same as the one that is used on a LINEGRP or TYPE EDIT= parameter.

If FPATH=YES is specified, the EDIT= keyword parameter specifies whether (UC) or not (ULC) the transaction is to be translated to uppercase before being presented to the edit/routing exit routine. Specification of a user edit routine is valid on a Fast Path potential transaction; this specification is used when the transaction is routed to IMS. It is invalid on a Fast Path exclusive transaction.

For input from LU 6.2 devices, the user edit exit routine DFSLUEE0 is called instead of the transaction input edit routine specified in EDIT.

Related Reading: For more information about DFSLUEE0, see *IMS/ESA Customization Guide*.

FPATH=

Specifies whether (YES, size) or not (NO) the transaction code is a potential candidate for Fast Path processing. FPATH=YES is effective only when specified on a TRANSACT statement that follows an APPLCTN statement that does **not** specify FPATH=YES; otherwise, the operand is ignored. FPATH=size, which determines the EMH buffer size required to run the transaction, overrides the EMHL execution parameter and implies FPATH=YES. The minimum specification for FPATH=size is 12; the maximum is 30720. The default is FPATH=NO.

Fast Path-potential transactions must be processed by a user edit/routing exit to determine whether the transaction is actually to be processed by IMS Fast Path. If it is to be processed by IMS Fast Path, the edit/routing exit routine associates the transaction with a routing code. This routing code identifies which Fast Path application program is to process the transaction.

If a Fast Path-potential transaction is also defined with transaction command security by the Security Maintenance utility (SMU), that transaction must be processed by IMS, not Fast Path. (That is, an automated operator interface transaction can also be a Fast Path potential transaction if you program your user edit/routing exit routine to route the transaction to IMS and not Fast Path.) The SMU cannot define a Fast Path-exclusive transaction as able to issue commands.

If FPATH=YES is specified during a MODBLKS system definition, Fast Path must have been previously defined for the online system.

INQ= or

INQUIRY=

Specifies whether (YES) or not (NO) this is an inquiry transaction. The default is

TRANSACT Macro

NO. You can specify this keyword either as INQ= or INQUIRY=. If INQ= (or INQUIRY=) YES is specified, you can also specify whether (RECOVER) or not (NORECOV) this transaction should be recovered during an IMS emergency or normal restart. The specification of INQ=(NO,NORECOV) is invalid. The default is RECOVER.

For IMS Fast Path transactions, RECOVER must be specified.

INQ=YES should be specified only for those transactions that, when entered, do not cause a change in any database. Programs are prohibited from issuing ISRT, DLET, or REPL calls to a database when scheduled to process a transaction defined as INQ=YES.⁵ The INQ operands are not position dependent.

If the SPA parameter is specified (indicating that the transaction is conversational), INQ=(YES,NORECOV) cannot be specified.

Because switched terminals signed on for the INQUIRY LTERM, (one that has not specified /IAM) 2741 terminals are not allowed to enter transactions that update a database, the INQ=YES parameter must be specified for transactions that are entered by these terminals.

If an attempt is made to enter a transaction that is not specified as INQ=YES from one of these terminals, the transaction is rejected. A message indicating that the update transaction cannot be processed is sent to the terminal that entered the transaction.

MAXRGN=

Limits the number of message processing program (MPP) regions that can be concurrently scheduled to process a transaction. When the number of MPP regions is not limited, one transaction might monopolize all available regions.

If you specify zero, or if zero is defaulted to, no limit is imposed. The maximum number you can specify is 255.

If you specify SERIAL=YES or SCHDTYP=SERIAL in the APPLCTN macro, omit the MAXRGN parameter or set it equal to 0.

For non-0 values, you cannot specify MAXRGN= unless you also specify PARLIM=.

MODE=

Specifies that database buffers are to be written to direct access (flushed) upon each request for a new message (SNGL) by the processing program, or upon program termination (MULT). The default is MULT. Conversational and WFI transactions must be defined as SNGL. SNGL is forced for WFI applications.

This operand affects emergency restart. When MODE=SNGL, emergency restart only reprocesses the last completed message, regardless of whether one or more messages are scheduled and processed by a single load of an application program. Otherwise, emergency restart reprocesses those messages that were scheduled and processed by the single load of an application program, since the time of the previous checkpoint. The number of messages processed depends upon when the last checkpoint is issued.

5. An application program cannot do an SQL INSERT, DELETE, or UPDATE when the IMS transaction is defined with INQ=YES.

The MODE= keyword parameters are checked on all IMS Fast Path–potential transactions for MODE=SNGL. If not specified, a warning diagnostic is issued.

MSGTYPE=

Specifies the type of transaction code (single or multiple segment), and whether the communication line from which the transaction is entered is to be held until a response is received. The MSGTYPE operands are not position dependent.

The transaction code can be single segment (SNGLSEG), or multiple segment (MULTSEG). It specifies the time at which an incoming message is considered complete and available to be routed to an application program for subsequent processing. The defaults are (MULTSEG, NONRESPONSE, 1).

If MSC-directed routing is used in a multiple IMS system configuration, IMS does not ensure that both the message and the transaction destined to process that message are either single segment or multiple segments.

The MSGTYPE= keyword parameters are checked on all IMS Fast Path–potential transactions for MSGTYPE=(SNGLSEG, RESPONSE). If not specified, a warning diagnostic is issued.

The first parameter of the MSGTYPE keyword specifies one of the following choices for number of segments:

MULTSEG

Specifies that the incoming message can be more than one segment in length. It is not eligible for scheduling to an application program until an end-of-message indication is received, or a complete message is created by MFS.

SNGLSEG

Specifies that the incoming message is one segment in length. It becomes eligible for scheduling when the terminal operator indicates end-of-segment.

The second parameter of the MSGTYPE keyword specifies one of the following response choices:

NONRESPONSE

Specifies that, for terminals specifying or accepting a default of OPTIONS=TRANRESP, input should not stop after this transaction is entered.

RESPONSE

Specifies that, for terminals specifying or accepting a default of OPTIONS=TRANRESP, no additional messages are to be allowed after this transaction is entered until this transaction sends a response message back to the terminal. Response mode can be forced or negated by individual terminal definition.

The third parameter of the MSGTYPE= keyword specifies the class to which this transaction code is to be assigned. This parameter must be a decimal number from 1 to 255. The default is 1.

Because the values associated with the class keyword can be changed using an /ASSIGN command, they will not be affected by an online change sequence of /MODIFY operator commands for an existing transaction, regardless of whether these values are altered during a MODBLKS system definition. The value specified must not exceed the MAXCLAS= value specified or accepted by

TRANSACT Macro

default on the IMSCTRL macro statement. Regardless of specification made or default taken, remote transaction codes are assigned a class of zero. If the transaction code class is specified in the APPLCTN macro, this parameter need not be specified. If the transaction code class is specified in both the APPLCTN and TRANSACT macros, the APPLCTN macro specification is ignored for this transaction. Define CPI transactions with a different message class from that used for non-CPI transactions. IMS handles all CPI transactions as priority zero within the transaction class.

An input transaction from a 2741 line, or 2740 nonstation control is treated as a RESPONSE-type transaction, without regard to the type specified for this operand. Note also that MSGTYPE=RESPONSE is ignored during online processing for all terminals that do not operate in response mode.

PARLIM=

Specifies the threshold value to be used when SCHDTYP=PARALLEL is specified in the preceding APPLCTN macro instruction. An additional region is scheduled whenever the current transaction enqueue count exceeds the PARLIM value multiplied by the number of regions currently scheduled for this transaction. If PARLIM is not specified, the default value of NONE is assumed, and IMS allows the transaction to be scheduled in only one region at a time. Valid values for PARLIM are from 0 to 32767.

PARLIM=0 indicates that any input message can cause a new region to be scheduled because the scheduling condition is always be met (the number of messages will be greater than zero).

If you specify PARLIM=0, you should specify a MAXRGN value to limit the number of regions that can be scheduled to process a particular transaction.

The value specified for PARLIM applies to message processing programs (MPPs) only; it is not supported for batch message processing programs (BMPs).

Because the value associated with the PARLIM= keyword can be changed using an /ASSIGN command, it is not affected by an online change sequence of operator commands for an existing transaction, regardless of whether it is altered during a MODBLKS system definition.

If you specify SERIAL=YES or SCHDTYP=SERIAL in the APPLCTN macro, omit the PARLIM parameter and accept the default.

In a Shared Queues environment (when SCHDTYPE=PARALLEL is specified on the previous APPLCTN macro), any PARLIM value other than NONE will cause a new region to be scheduled up the MAXRGN value. This is because there is no local message queue in which to determine the number of enqueued transactions.

PROCLIM=

Specifies the number of messages (count) of this transaction code a program can process in a single scheduling and the amount of time (in seconds) allowable to process a single transaction (or message). Batch Message Programs (BMPs) are unaffected by these settings.

The first parameter, *count*, can be changed using an */ASSIGN* command. Therefore, its value is not affected by an online change sequence of operator commands for an existing transaction, regardless of whether it was altered during a MODBLKS system definition.

The *count* specifies the maximum number of messages sent to the application program by the IMS control program for processing without reloading the application program. The *count* value can range from 0 through 65535. If 0 is coded, the maximum number of messages sent to the application is one and the application program is reloaded before receiving a subsequent message. Code the *count* value at 65535 if no limit is to be placed upon the number of messages processed at a single program load. Values 1 through 65535 are eligible for quick reschedule processing. The defaults for PROCLIM are 65535 and 65535.

The *second* parameter specifies a numeric value, in seconds, which can range from 1 to 65535.

If Fast Path is used, *second* specifies, for a given transaction code, the amount of time (in hundredths of seconds) the program is allowed to process a single transaction message. A valid specification ranges from 1 to 65535; if 65535 is specified, no time limit is placed on the application program. The time represents real time that elapses during transaction processing (not accumulated task time). Real time is used because the input terminal is in response mode and cannot enter another transaction until the response is sent. The *count* subparameter is ignored.

The number specified by *second* represents the maximum CPU time allowed for each message to be processed in the message processing region.

The *count* value assigned is used to determine how many messages an application program is allowed to process in a single scheduling cycle. When the application program requests, and receives, the number of messages indicated in the *count* value, any subsequent requests result in one of two things.

1. IMS indicates "no more messages exist" if any of the following conditions are true:
 - The region is not an MPP.
 - The currently scheduled mode is not MODE=SNGL.
 - Equal or higher-priority transactions are enqueued for the region.

IMS might, in fact, have other messages enqueued for the application program. It is the responsibility of the application program to terminate when it receives an indicator that no more messages are available. Termination of the application program makes the region it occupied available for rescheduling. This feature makes it possible for IMS to allow scheduling of higher-priority transactions that entered the system while the previous transactions were in process. In addition, if any equal-priority transactions are enqueued, they become eligible for scheduling on a first-in, first-out (FIFO) basis.

2. The region goes through quick reschedule and returns the next message to the application of all of the following conditions are true:
 - The region is an MPP.
 - The transaction is MODE=SNGL.

TRANSACT Macro

- No equal or higher transactions are enqueued.
- Messages are still enqueued for the application.

The *seconds* value controls application program looping. You are not required to optimize the *seconds* value for program-transaction execution time. However, the *seconds* time value assigned should not be less than the expected per-transaction execution time. If the scheduled application program exceeds the product of *seconds* and *count*, the application program abends. If an IMS STIMER value of 2 is specified on the DFSMPR macro, the region does not abend until completion of the DL/I call.

The application must not use MVS timer services, such as STIMER TASK, that override the IMS STIMER. IMS uses the IMS STIMER to time the execution of service message blocks. If an MVS TIMER is issued, it cancels out the IMS STIMER.

PRTY=

Specifies the values that determine the scheduling priority of this transaction. This priority also controls the priority of messages created by this transaction and sent to a destination in a remote system.

normal

The priority assigned to this transaction when the number of input transactions enqueued and waiting to be processed is less than the *limit count* value. The valid specification range is from 0 through 14. The default is 1.

limit

The priority to which this transaction is raised when the number of input transactions enqueued and waiting to be processed is equal to or greater than the *limit count* value. The valid specification range is from 0 through 14. The default is 1.

limit count

The number that, when compared to the number of input transactions queued and waiting to be processed, determines whether the *normal* or *limit* priority value is assigned to this transaction. The *limit count* value can range from 1 through 65535. The default is 65535.

When the limit priority is used, and the priority is raised to the specified limit priority value, the priority is not reduced to the normal priority until all messages enqueued for this transaction code are processed.

If you do not want the limit priority for this transaction, code equal values for the normal and limit priorities, and a limit count of 65535.

When a transaction is processed exclusively by a batch message program (BMP), code the normal and limit priorities as 0. The limit count value is ignored for a transaction processed by a BMP.

The APPLCTN macro statement forces the scheduling priority of all transaction codes associated with it to 0 if the program type is batch (PGMTYPE=BATCH on APPLCTN macro statement). However, a batch message processing region (BMP) can process transactions with scheduling priorities other than 0.

For remote transactions, the PRTY parameter determines the priority used to send the transaction to the processing system, which is termed the *MSC link message priority*. The three MSC link message priority groups are:

- Low
- Medium
- High

The low priority group consists of primary requests in the input terminal system. This group is assigned remote transaction priorities from 0 to 6. The medium group consists of secondary requests, responses, primary requests in an intermediate system, and primary requests in the input terminal system. This group is assigned a remote transaction priority of 7. The high group consists of primary requests in the input terminal system. Messages in this group are assigned remote transaction priorities from 8 to 14. Within each group, messages have a priority based on the current priority value of the transaction or remote transaction in the input terminal system for primary requests, and on the latest processing system for secondary requests and responses.

In an MSC configuration, the transaction priority determines the priority used to send messages inserted by this transaction across an MSC link. If the transaction inserts multiple messages to the same destination (for example, pages to a printer) and these messages must be sent in the order inserted, the normal and limit priority values should be the same. If the normal and limit priority values are not identical, messages inserted at a higher priority than previously inserted messages could arrive at their destination first. (This restriction does not apply to multiple segments of the same message.)

A transaction must have the same characteristics in all systems where it is defined. These characteristics include:

- Nonconversational/conversational
- SPA size if conversational
- Single-/multi-segment messages
- Noninquiry/inquiry
- Recoverable/nonrecoverable

ROUTING=

If MSC directed routing is used in a multiple IMS system configuration, specifies whether (YES) or not (NO) the application program processing a transaction is informed of the system which originated the transaction.

If ROUTING=YES, an MSNAME corresponding to a logical path back to the originating system is placed in the I/O PCB. If ROUTING=NO, the name of the originating LTERM is placed in the I/O PCB. The default is NO.

SCHD=

Specifies the scheduling option used for other transactions when this transaction cannot be scheduled for internal reasons (database intent or no more space in PSB pool or DMB pool to bring in needed blocks). Values include:

- 1 Schedule only transactions of equal or higher priority in the selected class. This is the default.
- 2 Schedule higher-priority transactions in the selected class.
- 3 Schedule any transaction in the selected class.
- 4 Skip to the next class and attempt to schedule the highest-priority transaction in that class.

TRANSACT Macro

A loop counter in MPP scheduling breaks potential scheduling loops between SCHD=1 or SCHD=2 and SCHD=3 transactions. If scheduling fails due to intent conflicts for a SCHD=1 or SCHD=2 transaction, the counter increments and the next logical transaction is selected based on the SCHD=parameter of the failed transaction.

Each time the scheduler fails to schedule a transaction for intent, the counter is checked. If the count is greater than five, the next class of transaction is selected and the counter reset. This processing method can cause a delay in the current class.

For example, a long-running BMP has update intent on a database. Several transactions that have update intent on the same database with SCHD=1 are entered into the IMS system. Transactions in the same class that do not reference the database are also added to the system. However, the first group of transactions fails for intent, and none of the transactions is scheduled until the BMP terminates.

After each failed transaction, the loop counter increments. After the counter exceeds 5, the next class is scheduled, bypassing the transactions that do not reference the database and causing delays in their processing.

To avoid these delays, place the second group of transactions into a separate class, or run the BMP job at a different time.

SEGNO=

Specifies the maximum number of application program output segments that are allowed into the message queues per Get Unique (GU) call from the application program. It must be specified as a decimal number from 0 through 65535. The default is 0. If the default specification of 0 is used, the number of segments is not checked by the online system at execution time.

Because the value associated with the SEGNO= keyword can be changed using an /ASSIGN command, it is not affected by an online change sequence of operator commands for an existing transaction, regardless of whether it is altered during a MODBLKS system definition.

SEGSIZE=

Specifies the maximum number of bytes allowed in any one output segment. It must be specified as a decimal number from 0 through 65535. The default is 0. If the default specification of 0 is used, the segment size is not checked by the online system at execution time.

Because the value associated with the SEGSIZE= keyword can be changed using an /ASSIGN command, it is not affected by an online change sequence of operator commands for an existing transaction, regardless of whether it is altered during a MODBLKS system definition.

The maximum output message segment to a LU 6.2 device is 32767. If a transaction is expected to send output to a LU 6.2 device, the SEGSIZE parameter should be no greater than 32767. However, this is not enforced during processing of the TRANSACT macro, because IMS cannot determine the device type for the message destination until output time.

SERIAL=

Forces serial processing of messages for a given transaction. When SERIAL=YES, U3303 pseudoabends do not cause the message to be placed

on the suspend queue but rather on the front of the transaction message queue, and the transaction is stopped with a USTOP.

The USTOP of the transaction is removed when the transaction or the class is started with a /START command .

The default for this keyword is NO, which means message processing is done as before with the messages placed on the suspend queue after a U3303 pseudoabend. Scheduling continues until repeated failures result in the transaction being stopped with a USTOP.

If you specify SERIAL=YES, PARLIM= and MAXRGN= must equal 0 (you can omit these parameters). If PARLIM= or MAXRGN= is not 0, SERIAL=YES is invalid, and the default of NO is assumed.

SPA=

Defines, by inclusion, that this transaction is a conversational transaction.

size

Specifies the size of the conversational scratchpad area (SPA). The size specified must be between 16 bytes and 32767 bytes inclusive.

STRUNC|RTRUNC

You can turn the truncated data option on (STRUNC) or off (RTRUNC).

If you specify SPA=STRUNC, IMS preserves all of the data in the SPA, even when a program switch is made to a transaction that is defined with a smaller SPA. The transaction with the smaller SPA does not see the truncated data, but when the transaction switches to a transaction with a larger SPA, the truncated data is used.

If you specify SPA=STRUNC, the truncated data is not preserved.

STRUNC is the default.

When a conversation initially starts, and when a program switches, the STRUNC|RTRUNC option is checked and set or reset as specified. When the option is set, it remains set for the life of the conversation, or until a program switch occurs to a transaction that specifies the option is to be reset.

When a program switch occurs, the truncated data option for the new transaction is first checked, and if specified (either STRUNC or RTRUNC), that specification is set for the conversation and is used for the SPA inserted into the output message. If the option is not specified for the new transaction, the option currently in effect for the conversation is used.

Restriction: The CORE, DASD, and FIXED operands can no longer be used. If you specify them, assembly errors occur.

SYSID=

In the multiple-IMS system configuration, specifies the system identification (SYSID) of the remote system (the system on which the application executes) and the SYSID of the local system (the originating system to which the responses are returned). The values specified must be from 1 through 2036. The remote SYSID specified must also be defined in an MSNAME macro statement, but the local SYSID can be defined in any or all of the MSNAME, TRANSACT, and APPLCTN macro statements.

TRANSACT Macro

If the SYSID parameter is specified in the APPLCTN macro statement, you need not specify the SYSID in the TRANSACT macro statement. If the SYSID is specified in both the APPLCTN and the TRANSACT macro statements, the APPLCTN specification is ignored.

The SYSID parameter is independent of the link type (CTC, MTM, VTAM) specified on the TYPE= keyword of the MSPLINK macro statement. Because the values associated with the SYSID= keyword can be changed using an /MSASSIGN command, they are not affected by an online change sequence of operator commands, regardless of whether they are altered during a MODBLKS system definition, except across a cold start of IMS.

A Fast Path–exclusive transaction cannot have SYSID= specified. To assign a remote transaction as local, the associated APPLCTN macro must be defined as local. This means that no SYSID= parameter can be specified for the associated APPLCTN macro.

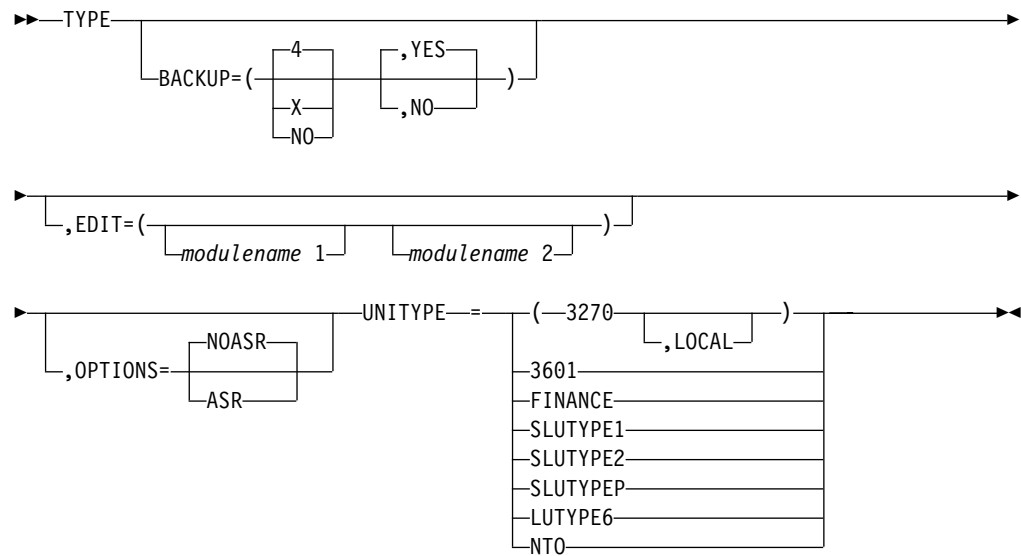
WFI

The positional parameter WFI specifies that this is a wait-for-input transaction. A message processing or batch processing application program that processes WFI transactions is scheduled and invoked normally. If the transaction to be processed is defined as WFI, the program is allowed to remain in main storage after it has processed the available input messages. The QC status code (no more messages) is returned to the program if the PROCLIM *count* is reached; a command is entered to change the status of the scheduled transaction, database, program or class; the /DBR, /DBD or /STA commands relating to the databases used by the transaction are entered, or IMS is terminated with a checkpoint shutdown. MODE=SNGL is forced when WFI is specified.

TYPE Macro

The TYPE macro statement defines the beginning of a set of communication terminals and logical terminal description macro statements which include TERMINAL and NAME. The TYPE macro statement begins a description of one set, that contains one or more terminals of the same type. TYPE defines terminals attached to IMS through VTAM. It is equivalent to the LINEGRP/LINE macro set used to define terminals attached to IMS by means other than VTAM.

In addition to the keyword parameters described below, any keyword parameter of the TERMINAL macro statement except EDIT, LTERM, NAME, COMPT, COMPT1 through COMPT4, and ADDR can be specified as a keyword operand of the TYPE macro statement. This provides default values for subsequent TERMINAL macro statements within this terminal description set.



Keyword Parameters

To find which parameters apply to your IMS configuration refer to Table 12 on page 32.

BACKUP=

Specifies (for XRF only) the control of automatic restart (BTAM) or session switching (VTAM) after takeover. Use only when HSB=YES is specified on the IMSCTRL macro.

X is a numeric integer from 1 to 7, inclusive, that specifies priority for reestablishing the session. The default is 4 when either keyword or parameter is omitted. NO suppresses session recovery of the terminal at takeover.

Although IMS sets priorities for VTAM requests, the active requests can be completed in any order because of internal VTAM conflicts and pacing.

The second parameter determines whether the backup system is to attempt to establish a backup session to the terminal when the active request establishes a session. When the entire keyword or this parameter is omitted, the default is YES. Specifying NO suppresses the establishment of a backup session.

EDIT=

modulename 1 specifies the one- to eight-byte name of a user-supplied physical terminal output edit routine for the terminals in this communication description set.

modulename 2 specifies the one- to eight-byte name of a user-supplied physical terminal input edit routine for the terminals in this communication description set.

This routine cannot be the same as the one that is used on a TRANSACT EDIT= parameter.

OPTIONS=

Specifies (for the Session Outage Notification facility only) automatic session restart (ASR) processing on all nodes of the same type. The default is NOASR.

TYPE Macro

The ASR and NOASR definitions on the TERMINAL macro override those definitions on the TYPE macro. However, the ASR option is not supported for nodes that can have XRF backup sessions.

Use the /DIS command to display the current ASR option, and the /CHANGE command to change it.

UNITYPE=

Specifies the terminal device type contained in this communication description set.

A terminal defined as either a 3770 or NTO is to be treated as a secondary logical unit type 1 terminal by IMS.

Terminals belonging to a Finance Communication System such as the 3600 Finance Communication System or the 4700 Finance Communication System can be specified as UNITYPE=FINANCE on the TYPE macro statement. Terminals belonging to the 3600 Finance Communication System can also be specified as UNITYPE=3601. The code generated to support the terminal is identical regardless of whether the terminal is specified as UNITYPE=3601 or UNITYPE=FINANCE.

VTAMPOOL Macro

▶▶—VTAMPOOL—◀◀

This macro, required for parallel session support, begins the definition of the LU 6.1 LTERM subpools. You must define VTAMPOOL if any LU 6.1 node uses dynamic LTERM allocation. LTERMs, defined as LU 6.1 nodes, can be a fixed set or may be dynamically allocated depending on the composition of the VTAM macro set. This macro initiates a hierarchy of macros, consisting of a VTAMPOOL statement and one or more subsequent SUBPOOL macros, each of which can optionally be followed by one or more NAME macros (if at least one SUBPOOL statement has an associated NAME statement). This macro set can occur anywhere in the VTAM macro set if it does not break into the TYPE—TERMINAL—NAME statement hierarchy. This macro has no operands.

To find if this macro is valid in your IMS configuration, refer to Table 12 on page 32.

All LU 6.1 SUBPOOLS can be defined under a single VTAMPOOL statement or under multiple VTAMPOOL statements. The only purpose of defining LU 6 SUBPOOLS using multiple VTAMPOOL statements is to aid in documenting the use of each LU 6 SUBPOOL.

Chapter 4. Procedures

This chapter provides the following information for each of the IMS Sysgen-supplied procedures:

- A sample of each procedure
- Lists of valid parameters and DD statements for each procedure

The parameters are listed alphabetically and descriptions of them begin on page 244. The DD statements are listed alphabetically and descriptions of them begin on page 279.

In this Chapter:

- “Using IMS Sysgen-Supplied Procedures” on page 191
- “Specifying the DFSRESLB DD Statement in IMS Batch Procedures” on page 192
- “DBBBATCH Procedure” on page 192
- “DBC Procedure” on page 195
- “DBRC Procedure” on page 200
- “DCC Procedure” on page 201
- “DFSMPR Procedure” on page 211
- “DLIBATCH Procedure” on page 212
- “DLISAS Procedure” on page 214
- “DXRJPROC Procedure” on page 216
- “FDR Procedure” on page 217
- “FPUTIL Procedure” on page 221
- “IMS Procedure” on page 221
- “IMSBATCH Procedure” on page 232
- “IMSCOBGO Procedure” on page 233
- “IMSCOBOL Procedure” on page 235
- “IMSFP Procedure” on page 236
- “IMSMSG Job” on page 237
- “IMSPLI Procedure” on page 238
- “IMSPLIGO Procedure” on page 239
- “IMSRDR Procedure” on page 241
- “RDIBATCH Procedure” on page 242
- “DL/I Interfaces” on page 243
- “Parameter Descriptions” on page 244
- “DD Statement Descriptions” on page 279

The following table shows all the IMS SYSGEN-supplied cataloged procedures, control statements, and jobs. Unless noted otherwise, all items in the first column of the table are procedures.

IMS Sysgen-Supplied Procedures

Table 18. IMS SYSGEN-Supplied Cataloged Procedures and the Environments in Which They Apply

Name	DB Batch	TM Batch	DBCTL	DB/DC	DCCTL	Description	Page
DBBBATCH	X	X				A one-step procedure for offline DL/I batch processing regions using IMS.ACBLIB.	192
DBC			X			An online execution procedure to initialize the DBCTL environment.	195
DBRC			X	X	X	A procedure to execute the DBRC address space.	200
DCC					X	An online execution procedure to initialize the DCCTL environment.	201
DFSMR				X	X	A procedure to execute an IMS message processing address space.	211
DLIBATCH	X	X				A one-step procedure for an offline DL/I batch processing program using PSB and DBD libraries.	212
DLISAS			X	X		A procedure to execute the DL/I address space.	214
DXRJPROC	X		X	X		A procedure that defines the Internal Resource Lock Manager and its use to the system.	216
FDR			X	X		A procedure to execute an FDBR region.	217
FPUTIL			X	X		A procedure to execute the Fast Path utility program with DEDBs online.	221
IMS				X		An online execution procedure to initialize the DB/DC environment.	221
IMSBATCH			X	X	X	A procedure to execute an IMS online batch message processing address space.	232
IMSCOBGO	X	X				A three-step compile, link-edit, and go procedure combining the IMSCOBOL procedure with an execution step for a stand-alone DL/I batch address space.	233
CBLTDLI ²	X	X	X	X	X	Control statements necessary to establish a COBOL-to-DL/I interface.	243
IMSCOBOL	X	X	X	X	X	A two-step compile and link-edit procedure for IMS applications written in COBOL.	235
IMSFP				X	X	A procedure for executing a Fast Path application program.	236
IMSMSG ¹				X	X	A job to execute an IMS message processing program.	419
IMSPLI	X	X	X	X	X	A two-step compile and link-edit procedure for IMS applications written in PL/I.	238
IMSPLIGO	X	X				A three-step compile, link-edit, and go procedure combining the IMSPLI procedure with an execution step for a stand-alone DL/I batch processing address space.	239
PLITDLI ²	X	X	X	X	X	Control statements necessary to establish a PL/I-to-DL/I interface.	243

IMS Sysgen-Supplied Procedures

Table 18. IMS SYSGEN-Supplied Cataloged Procedures and the Environments in Which They Apply (continued)

Name	DB Batch	TM Batch	DBCTL	DB/DC	DCCTL	Description	Page
IMSRDR			X	X	X	A DASD read procedure to read an IMSMSG job into the operating system job stream from direct access storage devices.	241
RDIBATCH	X		X	X		A procedure to help maintain database availability in failure situations.	242

Notes:

¹ Job

² Control Statement

Related Reading: The information concerning the following supplied cataloged procedures, control statements, and jobs is found in the following books:

IMS/ESA Utilities Reference: Transaction Manager

DFSMREC
DFSWTnnn
FMTCPY
IMSMSV
IMSWTnnn
INITMOD
MFSBACK
MFSBTCH1
MFSBTCH2
MFSUCT
MFSREST
MFSRVC
MFSTEST
MFSUTL
OLCUTL
REFCPY

IMS/ESA Utilities Reference: System

ACBGEN
DBDGEN
DFSACBCP
DFSMREC
IMSDALOC
PSBGEN
SECURITY

Using IMS Sysgen-Supplied Procedures

This section describes some general considerations on creating, modifying, and storing IMS sysgen-supplied procedures. The generated procedures accommodate only MVS/ESA.

Storing Procedures

The DBC, DBRC, DCC, DLISAS, IMS, and IMSRDR procedures should be stored in SYS1.PROCLIB. The JCL supplied in the IMSMSG and IMSWTnnn members of IMS.PROCLIB must be modified as your installation requires and stored in IMS.JOBS. The other IMS-supplied procedures can be left in IMS.PROCLIB, where they are placed during IMS system definition.

If PROCLIB=YES is specified when preparing the IMSGEN system definition macro statement, certain procedures and the jobs IMSMSG and IMSWTnnn are dynamically created and placed in IMS.PROCLIB. The created jobs and procedures should be examined carefully to determine whether the JCL is generated as you require. These procedures might not apply to all applications, but they can be used

IMS Sysgen-Supplied Procedures

as guidelines for user-generated account oriented procedures. Depending on the type of system being defined, your procedure library members can be a subset of the complete IMS procedure library that is presented here.

IMS conforms to MVS rules for data set authorization. If an IMS job step is to run authorized, all libraries to be used in that job step must be authorized. To run an IMS batch region as unauthorized, a nonauthorized library must be concatenated to IMS.RESLIB.

Allocating Terminal Devices in an Online IMS System

If an online IMS system has been defined, particular attention should be devoted to the terminal device allocation generated within the IMS procedure. A list of terminal addresses and logical and physical terminals is printed by stage 1 of IMS system definition. Examples of the procedure jobs in this chapter show the contents of the members as they are supplied. No card column image is intended. When coding your own procedures, follow JCL and Assembler language coding practices.

Specifying the DFSRESLB DD Statement in IMS Batch Procedures

Certain batch procedures must load the IMS SVC modules, all of which must reside in an authorized library. The DFSRESLB DD statement specifies the library that contains these modules. All libraries specified by this statement must be authorized through the Authorized Program Facility (APF).

When you include the DFSRESLB DD statement in a batch procedure, all IMS SVC modules are loaded from the specified library. If this library does not contain the required modules, IMS does not search any other library for the modules and then terminates.

When you do not include the DFSRESLB DD statement in a batch procedure, IMS searches the standard default libraries for the modules (JOBLIB/STEPLIB, followed by LINKLIB). If the IMS SVC modules do not reside in an authorized library, IMS terminates.

In addition to the DFSRESLB DD statement, the EXEC parameter SRCH, which is allowed on selected batch procedures, also affects the library search order. When SRCH=1, IMS searches the link pack area and directories prior to searching any other library. When SRCH=0 and the DFSRESLB DD statement is specified, IMS searches only the specified library and does not search the link pack area.

The following modules must be loaded from an authorized library during the IMS batch initialization:

DFSABND0	DFSMODU0
DFSAOSF0	DFSRTM00
DFSAOS70	DFSSPF00
DFSBC000	DFSSTM00
DFSCBT10	DFSTOPR0
DFSFSWA0	DFSVC100
DFSKPXT0	DFSVC110

DBBBATCH Procedure

The DBBBATCH procedure is a one-step procedure for running an offline DL/I batch processing region using IMS.ACLIB.

The following restrictions apply when you run DL/I batch jobs in an RSR environment:

Restrictions:

- The batch-only databases that are identified to be tracked must be included in the system definition (DATABASE macro) for the tracking subsystem.
- The default GSG and TMI names used by batch jobs are specified in the IMSCTRL macro, but they can be overridden in the DLIBATCH procedure.
- Batch jobs do not use the DFSRSRxx PROCLIB member, and are therefore limited to the default VTAM modename (TMDEFLT).

Procedure Description

The following procedure runs an offline DL/I batch processing region using IMS.ACBLIB.

If VSAM databases are used, see “Specifying IMS Buffer Pools” on page 378.

```
//      PROC MBR=TEMPNAME,PSB=,BUF=7,
//          SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,
//          SRCH=0,CKPTID=,MON=N,LOGA=0,FMT0=T,
//          IMSID=,SWAP=,DBRC=,IRLM=,IRLMNM=,
//          BKO=N,IOB=,SSM=,APARM=,
//          RGN=2048K,
//          SOUT=A,LOGT=2400,
//          SYS=,
//          SYS2=,LOCKMAX=,
//          GSGNAME=,TMINAME=
//G      EXEC PGM=DFSRR00,REGION=&RGN,
//          PARM=(DBB,&MBR,&PSB,&BUF,
//          &SPIE&TEST&EXCPVR&RST,&PRLD,
//          &SRCH,&CKPTID,&MON,&LOGA,&FMT0,
//          &IMSID,&SWAP,&DBRC,&IRLM,&IRLMNM,
//          &BKO,&IOB,&SSM,
//          '&APARM',&LOCKMAX,
//          &GSGNAME,&TMINAME)
//STEPLIB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
//          DD DSN=IMS.&SYS2.PGMLIB,DISP=SHR
//DFSRESLB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
//IMSACBA DD DSN=IMS.&SYS2.ACBLIBA,DISP=SHR
//IMSACBB DD DSN=IMS.&SYS2.ACBLIBB,DISP=SHR
//MODSTAT DD DSN=IMS.&SYS2.MODSTAT,DISP=SHR
//MODSTAT2 DD DSN=IMS.&SYS2.MODSTAT2,DISP=SHR
//PROCLIB DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR
//IEFRDER DD DSN=IMSLOG,DISP=(,KEEP),VOL=(,,99),
//          UNIT=(&LOGT,,DEFER),
//          DCB=(RECFM=VB,BLKSIZE=1920,
//          LRECL=1916,BUFNO=2)
//IEFRDER2 DD DSN=IMSLOG2,DISP=(,KEEP),VOL=(,,99),
//          UNIT=(&LOGT,,DEFER,SEP=IEFRDER),
//          DCB=(RECFM=VB,BLKSIZE=1920,
//          LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&SOUT,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=605),
//          SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON  DD DUMMY
```

Notes:

- Parameters in parentheses are positional.
- For an XRF-capable system only.
- The IEFRDER statement is not required if the job does not declare database update intent.

DBBBATCH Procedure

- IEFRDER2 is optional.

For a job step declaring database update intent, DD DUMMY can be specified, if the job step is not using DBRC. This is valid where an image copy of the database is taken before the update job step.

Log initialization calculates the smallest value necessary for logical record length. If the JCL logical record length value is larger than the calculated value, the JCL value is used; otherwise, log initialization uses the calculated value for logical record length and adds 4 for the block size.

If multiple volumes are required for the system log, a volume count value should be specified in the VOL parameter of the DD statement.

Related Reading: For information on the number of volumes required per DD statement, see *MVS/ESA Job Control Language User's Guide*.

If the IBM 3480 tape drive is used for the IMS log data set, IMS forces tape write mode (DCB=OPTCD=W). The default on the 3480 is to buffer the write so that IMS cannot detect when the write is actually performed. If a power failure occurs after a log record is written to the 3480, and the database is updated but the log record is not yet written to tape, database integrity is lost. Tape write mode is forced for the log in batch and GSAM data sets.

DBBBATCH Parameters

The following parameters are valid for the DBBBATCH procedure. See "Parameter Descriptions" on page 244.

APARM=	MBR=
BKO=	MON=
BUF=	PRLD=
CKPTID=	PSB= ²
DBRC=	RGN=
EXCPVR=	RST=
FMTO=	SOUT=
GSGNAME=	SPIE=
IMSID=	SRCH=
IOB= ¹	SSM=
IRLM=	SWAP=
IRLMNM=	SYS=
LOCKMAX=	SYS2=
LOGA= ¹	TEST=
LOGT=	TMINAME=

Notes:

1. This parameter is no longer used and is ignored if specified.
2. This parameter is optional.

DBBBATCH DD Statements

The following DD statements are valid for the DBBBatch procedure.

In addition to the following DD statements, add statements for data sets representing IMS databases that are not to be dynamically allocated.

See “DD Statement Descriptions” on page 279.

DFSRESLB DD	IMS LOGR DD
DFSSTAT DD ¹	MODSTAT DD
DFSVSAMP DD	MODSTAT2 DD
IEFRDER DD	PROCLIB DD
IEFRDER2 DD ¹	RECONn DD ¹
IMSACBA DD	STEPLIB DD
IMSACBB DD	SYSUDUMP DD ¹
IMS DD	

Note:¹ This statement is optional.

DBC Procedure

Procedure Description

The following procedure executes an IMS DBCTL control region.

If DL/I databases are used, a member DFSVSMxx is built in IMS.PROCLIB. See “Specifying IMS Buffer Pools” on page 378.

```
//          PROC RGN=2000K,SOUT=A,DPTY='(14,15)',
//          SYS=,SYS2=,
//          RGSUF=DBC,PARM1=,PARM2=
//IEFPROC EXEC PGM=DFSMVRC0,DPRTY=&DPTY,REGION=&RGN,
//          PARM='DBC,&RGSUF,&PARM1,&PARM2'
//*
//*
//* THE MEANING AND MAXIMUM SIZE OF EACH PARAMETER
//* IS AS FOLLOWS:
//*****
//* RGSUF XXX EXEC PARM DEFAULT BLOCK SUFFIX FOR
//* MEMBER DFSPBXXX.
//*****
//*
//* PARM1 , PARM2 PARAMETERS BOTH ARE USED TO SPECIFY
//* CHARACTER STRINGS THAT CONTAIN IMS KEYWORD
//* PARAMETERS. I.E. PARM1='AUTO=Y,PST=222,RES=Y'
//*
//* ALL OF THE VALID DBCTL KEYWORD PARAMETERS
//* ARE DESCRIBED BELOW
//*****
//*
//***** CONTROL REGION SPECIFICATIONS *****
//* RES X BLOCK RESIDENT (N = NO, Y = YES)
//* PST XXX NUMBER OF PST'S PERMANENTLY ALLOC
//* MAXPST XXX MAXIMUM NUMBER OF PST'S
//* SRCH X MODULE SEARCH INDICATOR FOR DIRECTED LOAD
//*
//* 0 = STANDARD SEARCH
//* 1 = SEARCH JPA AND LPA BEFORE PDS
//* FMTO T = ONLINE FORMATTED DUMP WITH
//* STORAGE IMAGE DELETIONS.
//* TERM=NO SDUMPS ALLOWED.
//*
//* P = FULL ONLINE FORMATTED DUMP.
//* TERM=NO SDUMPS ALLOWED.
//*
//* F = FULL ONLINE FORMATTED DUMP.
//* TERM=NO SDUMPS NOT ALLOWED.
//*
//* N = NO FORMATTED DUMP, NO OFFLINE
```

DBC Procedure

```

/**          DUMP. TERM=NO SDUMPS ALLOWED.
/**          Z = NO FORMATTED DUMP, NO OFFLINE
/**          DUMP. TERM=NO SDUMPS NOT
/**          ALLOWED.
/** (DEFAULT) D = OFFLINE DUMP, OR ONLINE FORMAT-
/**          TED DUMP WITH STORAGE IMAGE
/**          DELETIONS IF OFFLINE DUMPING
/**          FAILS. TERM=NO SDUMPS ALLOWED.
/**          X = OFFLINE DUMP, OR ONLINE FORMAT-
/**          TED DUMP WITH STORAGE IMAGE
/**          DELETIONS IF OFFLINE DUMPING
/**          FAILS. TERM=NO SDUMPS NOT
/**          ALLOWED.
/**          M = OFFLINE DUMP, ONLINE IMS DUMP
/**          FORMATTING NOT PERMITTED
/**          TERM=NO SDUMPS ALLOWED.
/**          R = OFFLINE DUMP, ONLINE IMS DUMP
/**          FORMATTING NOT PERMITTED
/**          TERM=NO SDUMPS NOT ALLOWED.
/** AUTO X Y = AUTOMATIC RESTART DESIRED
/**          N = NO AUTOMATIC RESTART
/** IMSID XXXX IMS SUBSYSTEM IDENTIFIER
/** ISIS X 0 = NO RESOURCE ACCESS SECURITY
/**          1 = RACF RESOURCE ACCESS SECURITY
/**          2 = USER RESOURCE ACCESS SECURITY
/** ARMRST X Y = ALLOW MVS ARM TO RESTART
/**          N = ARM NOT RESTART IMS
/** IRLM X Y = YES, N = NO
/** IRLMNM XXXX IRLM SUBSYSTEM NAME
/** SSM XXXX EXT SUBSYSTEM PROCLIB MEMBER ID
/** WADS X SINGLE OR DUAL WADS,S=SINGLE,D=DUAL
/** ARC XX AUTOMATIC ARCHIVE.
/**          0 = NOT AUTOMATIC
/**          1-99 = AUTOMATIC
/** UHASH XXXXXXXX USER HASH MODULE NAME
/** DBRCNM XXXXXXXX DBRC PROCLIB MEMBER NAME
/** DLINM XXXXXXXX DL/I PROCLIB MEMBER NAME
/** PRDR XXXXXXXX IMSRDR PROCLIB MEMBER NAME
/** DBRSE XXXXXXXX 8 CHAR DBCTL RSENAME
/** CRC X COMMAND RECOGNITION CHARACTER
/** CMDMCS X COMMAND SECURITY OPTION
/**          R=RACF COMMAND SECURITY
/**          C=DFSCCMD0 COMMAND SECURITY
/**          B=RACF AND DFSCCMD0 CMD SEC
/** PREMSG X PREFIX MESSAGE OPTION
/**          N=NO DFS000I PREFIX
/**          Y=DEFAULT, DFS000I PREFIX
/** PIMAX XXXXXX ENQ/DEQ POOL MAXIMUM BYTES
/** PIINCR XXXXXX ENQ/DEQ POOL INCREMENT
/** AOIS X ICMD SECURITY OPTION
/** YEAR4 X N = 2-DIGIT DATE
/**          Y = 4-DIGIT DATE
/**
/******* FDR PARAMETER *****
/**
/** FDRMBR XX SUFFIX FOR FDR MEMBER IN
/**          IMS.PROCLIB
/**
/******* FAST PATH PARAMETERS *****
/**
/** BSIZ XXX DATA BASE BUFFER SIZE

```

```

/** OTHR   XXX   NUMBER OF OUTPUT THREADS
/** DBFX   XXX   SYSTEM ALLOCATION OF DATA BASE BUFFERS TO BE
/**                   FIXED AT START OF 1ST FAST PATH DEP REGION
/** DBBF   XXX   NUMBER OF DATABASE BUFFERS
/** LGNR   XX    NUMBER OF LOG ENTRIES IN DEDB BUFFERHEADER
/**
/******* RSR PARAMETERS *****
/**
/** RSRMBR XX    SUFFIX FOR RSR MEMBER
/** TRACK   XXX   NO = NO RECOVERY TRACKING DONE
/**                   RLT = RECOVERY TRACKING DONE
/**                   DLT = DATABASE TRACKING DONE
/**
/******* STORAGE POOL VALUES IN K, M OR G *****
/** DMB     XXXXXX DMB POOL SIZE
/** CIOP    XXXXXX CIOP POOL UPPER LIMIT
/** WKAP    XXXXXX WORKING STORAGE BUFFER POOL SIZE
/** PSBW    XXXXXX PSB WORK POOL SIZE
/** DBWP    XXXXXX DATABASE WORK POOL SIZE
/** CSAPSB  XXXXXX DLISAS: CSA PSB POOL SIZE
/** DLIPSB  XXXXXX DLISAS: DLI PSB POOL SIZE
/** EPCB    XXXXXX EPCB POOL SIZE
/** FWP     XXXXXX FP WORK POOL UPPER LIMIT
/** AOIP    XXXXXX AOI POOL UPPER LIMIT
/**
/******* MEMBER SUFFIXES *****
/**
/** SUF     X     LAST CHARACTER OF CTL PROGRAM LOAD
/**                   MODULE MEMBER NAME
/** FIX     XX    2 CHARACTER FIX PROCEDURE MODULE SUFFIX
/** PRLD    XX    2 CHARACTER PROCLIB MEMBER SUFFIX FOR PRELOAD
/** VSPEC   XX    2 CHARACTER BUFFER POOL SPEC MODULE SUFFIX
/** SPM     XX    STORAGE POOL OPTIONS (DFSSPMXX)
/*******
/**
/**STEPLIB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
/**PROCLIB DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR
/**
/*******
/** IN ORDER TO START A DEPENDENT REGION, MODIFIED
/** START-UP JCL IS WRITTEN FROM INTERNAL STORAGE TO
/** THE INTERNAL READER.
/**
/**IMSIRD  DD SYSOUT=(A,INTRDR)
/**
/******* DASD LOGGING DD CARDS *****
/** THE FOLLOWING DD CARDS DESCRIBE THE DASD LOGGING
/** OLDS AND WADS. THESE CARDS ARE FOR EXAMPLE ONLY.
/** ALL OLDS AND WADS DATA SETS MAY BE DYNAMICALLY
/** ALLOCATED. DD CARDS ARE NOT REQUIRED.
/** THE OLDS AND WADS TO BE USED DURING STARTUP MUST
/** BE SPECIFIED VIA OLDSDEF AND WADSDEF CONTROL
/** STATEMENTS IN THE DFSVSMXX MEMBER OF IMS PROCLIB.
/** THE ACTUAL SELECTION OF OLDS AND WADS MUST BE
/** TAILORED TO INSTALLATION REQUIREMENTS. THE OLDS
/** AND WADS MUST BE PREDEFINED BY A SET UP JOB.
/** THE BLOCK SIZE OF ALL OLDS MUST BE THE SAME.
/** THE BLOCK SIZE AND DEVICE TYPE OF ALL WADS MUST
/** BE THE SAME. AT LEAST 3 PRIMARY OLDS AND 1 WADS
/** MUST BE AVAILABLE FOR STARTUP. THE BLOCK SIZE
/** SHOULD NOT BE SPECIFIED IN THIS JCL. THE LOGGER

```

DBC Procedure

```
/** WILL GET THE BLOCK SIZE FROM THE VTOC.
/**
//DFSOLP00 DD DSN=IMS.&SYS.OLP00,DISP=SHR
//DFSOLP01 DD DSN=IMS.&SYS.OLP01,DISP=SHR
//DFSOLP02 DD DSN=IMS.&SYS.OLP02,DISP=SHR
//DFSOLP03 DD DSN=IMS.&SYS.OLP03,DISP=SHR
//DFSOLP04 DD DSN=IMS.&SYS.OLP04,DISP=SHR
//DFSOLP05 DD DSN=IMS.&SYS.OLP05,DISP=SHR
/**
//DFSOLS00 DD DSN=IMS.&SYS.OLS00,DISP=SHR
//DFSOLS01 DD DSN=IMS.&SYS.OLS01,DISP=SHR
//DFSOLS02 DD DSN=IMS.&SYS.OLS02,DISP=SHR
//DFSOLS03 DD DSN=IMS.&SYS.OLS03,DISP=SHR
//DFSOLS04 DD DSN=IMS.&SYS.OLS04,DISP=SHR
//DFSOLS05 DD DSN=IMS.&SYS.OLS05,DISP=SHR
/**
//DFSWADS0 DD DSN=IMS.&SYS.WADS0,DISP=SHR
//DFSWADS1 DD DSN=IMS.&SYS.WADS1,DISP=SHR
/**
//IMSACBA DD DSN=IMS.&SYS2.ACBLIBA,DISP=SHR
//IMSACBB DD DSN=IMS.&SYS2.ACBLIBB,DISP=SHR
//MODBLKSA DD DSN=IMS.&SYS2.MODBLKSA,DISP=SHR
//MODBLKSB DD DSN=IMS.&SYS2.MODBLKSB,DISP=SHR
//MODSTAT DD DSN=IMS.&SYS.MODSTAT,DISP=SHR
//***** SYSTEM REQUIRED DD CARDS *****
/**
//SYSUDUMP DD SYSOUT=&SOUT,
//          DCB=(LRECL=125,RECFM=FBA,BLKSIZE=3129),
//          SPACE=(6050,300,,ROUND)
//IMSRDS DD DSN=IMS.&SYS.RDS,DISP=SHR
//MATRIXA DD DSN=IMS.&SYS2.MATRIXA,DISP=SHR
//MATRIXB DD DSN=IMS.&SYS2.MATRIXB,DISP=SHR
//PRINTDD DD SYSOUT=&SOUT
/**
//***** EXTERNAL SUBSYSTEM DD CARDS *****
/**
/** USER MAY OPTIONALLY ADD THE DFSESL DD CARD
/** FOR EXTERNAL SUBSYSTEM CONNECTION.
/**
//***** DATA BASE DD CARDS *****
/**
/** USER MAY OPTIONALLY SUPPLY THE DD STATEMENTS
/** FOR THE ON-LINE DATA BASES TO BE
/** INSERTED HERE PRIOR TO ATTEMPTING
/** AN ON-LINE SYSTEM EXECUTION USING
/** THIS PROCEDURE.
/** IF NO DD STATEMENTS ARE SUPPLIED FOR
/** A DATA BASE, IMS ASSUMES THAT THIS
/** DATA BASE HAS BEEN DESCRIBED THROUGH
/** THE DFSMDA MACRO.
/** IF THE USER WILL BE EXECUTING WITH THE DL/I
/** SAS OPTION, THESE DD STATEMENTS SHOULD BE ADDED
/** TO THE DLISAS PROCLIB MEMBER OR DESCRIBED
/** THROUGH THE DFSMDA MACRO.
/**
```

Note:¹ DEDB areas must be allocated in the DBCTL control region. One of the following must occur:

- The DD statements be in the control region JCL.

- The dynamic allocation members must be in a library, as addressed by JOBLIB and STEPLIB.

DBCTL execution parameters can be specified on the DBC EXECUTE statement, in the DBC procedure, in the defaults member DFSPBDBC, or the parameter default values specified during system generation can be used. Following is a complete list of the parameters that apply to a DBCTL DBC region.

The DBC procedure is used “as is” for both regions in a DBCTL stand-by configuration.

DBC Parameters

The following parameters are valid for the DBC procedure. See “Parameter Descriptions” on page 244.

AOIP=	IRLMNM=
AOIS=	ISIS=
ARC=	LGNR=
ARMRST=	MAXPST=
AUTO=	OTHR=
BSIZ=	PIINCR=
CIOP=	PIMAX=
CMDMCS=	PRDR=
CORE=	PREMSG=
CRC=	PRLD=
CSAPSB=	PSBW=
DBBF=	PST=
DBRCNM=	RES=
DBFX=	RGUF=
DBRSE=	RSRMBR=
DBWP=	SPM=
DLINM=	SRCH=
DLIPSB=	SSM=
DMB=	SUF=
EPCB=	TRACK=
FIX=	UHASH=
FMTO=	USERVAR=
FPWP=	VSPEC=
FDRMBR=	WADS=
IMSID=	WKAP=
IOB=	YEAR4=
IRLM=	

DBC DD Statements

Valid DD statements for the DBC procedure are listed below.

In addition to the following DD statements, add statements for data sets representing IMS databases that are not to be dynamically allocated. See “DD Statement Descriptions” on page 279.

DFSOLPnn DD	MATRIXB DD
DFSOLSNnn DD	MODBLKSA DD
DFSWADSnn DD	MODBLKSB DD
IMSACBA DD	MODSTAT DD

DBC Procedure

IMSACBB DD
IMSIRD DD
IMSMON DD
IMSRDS DD
MATRIXA DD

PRINTDD DD
PROCLIB DD
STEPLIB DD
SYSUDUMP DD

DBRC Procedure

Procedure Description

The following procedure initializes the DBRC subordinate address space and is executed by IMS via an MVS START command during control region initialization.

```
//      PROC RGN=2048K,DPTY='(14,15)',SOUT=A,  
//      IMSID=SYS3,SYS2=  
//IEFPROC EXEC PGM=DFSMVRC0,REGION=&RGN,  
//      DPTY=&DPTY,PARM=(DRC,&IMSID)  
//*****  
//*  
//STEPLIB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR  
//PROCLIB DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR  
//JCLOUT DD SYSOUT=(A,INTRDR)  
//JCLPDS DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR  
//SYSUDUMP DD SYSOUT=&SOUT.  
//SYSABEND DD SYSOUT=&SOUT.  
//*  
//***** DBRC RECON DD CARDS *****  
//*  
//* USER MAY OPTIONALLY SUPPLY THE DD CARDS  
//* REQUIRED FOR THE DBRC RECON DATA SET.  
//* IF NO DD STATEMENTS ARE SUPPLIED FOR RECON  
//* DATASETS, IMS ASSUMES THAT THE DATASETS  
//* HAVE BEEN DESCRIBED THROUGH THE DFSMDA MACRO.  
//*
```

IMS system generation stores the cataloged procedure for the DBRC subordinate address space in the IMS.PROCLIB member specified in the DBRCNM=*member* parameter on the IMSCTRL macro. This *member* must be copied to SYS1.PROCLIB.

The IMS control region automatically initiates the DBRC address space procedure. The PDS member name that IMS uses for the DBRC procedure is defined by:

DBRCNM= DBRC default name
DBRCNM= Member in SYSGEN IMSCTRL macro, overrides default
DBRCNM= Member defined in DFSPBxxx, overrides SYSGEN value
DBRCNM= member in JCL exec parm, overrides PB member value

Default Process: An MVS START command is issued for DBRCNM. The MVS START command is coded to override the PARM=(DRC,&IMSID) procedure statement. The override defines the IMSID of the subsystem issuing the START command. This allows a generic procedure to be used by multiple IMS control regions.

IMS does not issue the IMS READY message until the DBRC address space successfully initiates. If the DBRC procedure started by the control region is in error, correct the procedure and restart it.

When the DBRC address space connects to the IMS control region, the DBRC procedure name is compared with DBRCNM and if they are not the same, the DBRC address space abends.

Alternate Process: The user can supply a START command to be issued instead of the internally defined START. The START command is supplied in member =DBRCNM of a PDS defined in //PROCLIB in the control region JCL. If columns 1-5 are not "START", message DFS1930I is issued and the internal START command is issued as if the DBRCNM member was not found in PROCLIB.

Example: Following are examples of START commands a user can specify:

- START IMSRDR,MBR=DBRC: the PDS member specified is read in as a job through the MVS internal reader. It starts as a job rather than as a started task.
- START DBRCOA,PARM=(DRC,IMSA): DBRC executes as a started task. DBRCOA connects to IMSA.

When the DBRC address space connects to the IMS control region, the DBRC procedure name is compared with the DBRCNM= specification. If they are not the same, the DBRC address space abends.

DBRC Parameters

The following parameters are valid for the DBRC procedure. See "Parameter Descriptions" on page 244.

DPRTY=	SOUT=
IMSID=	SYS2=
RGN=	

DBRC DD Statements

The following DD statements are valid for the DBRC procedure. See "DD Statement Descriptions" on page 279.

JCLOUT DD	STEPLIB DD
JCLPDS DD	SYSABEND DD
PROCLIB DD	SYSUDUMP DD
RECONn DD ¹	

Note: ¹ This DD statement is optional.

DCC Procedure

Procedure Description

The following procedure executes an IMS DCCTL address space.

```
//      PROC  RGN=3000K,SOUT=A,DPTY='(14,15)',
//          SYS=,SYS1=,SYS2=,
//          LOGT=2400,
//          RGSUF=DCC,PARM1=,PARM2=
//IEFPROC EXEC PGM=DFSMVRC0,DPRTY=&DPTY,REGION=&RGN,
//          PARM='DCC,&RGSUF,&PARM1,&PARM2'
//*
//*****
//*
//* THE DESCRIPTION AND CHARACTERISTICS OF EACH *
```

DCC Procedure

```
/** PARAMETER IS DEFINED AS FOLLOWS:          *
/**                                           *
/*******
/**
/******* CONTROL REGION SPECIFICATIONS *****
/**
/** RGSUF   XXX   EXEC PARM DEFAULT BLOCK SUFFIX FOR
/**                MEMBER DFSPBXXX.
/*******
/**
/** PARM1 , PARM2 PARAMETERS BOTH ARE USED TO SPECIFY
/** CHARACTER STRINGS THAT CONTAIN IMS KEYWORD
/** PARAMETERS. I.E. PARM1='AUTO=Y,PST=222,RES=Y'
/**
/** ALL OF THE VALID DCCTL KEYWORD PARAMETERS
/** ARE DESCRIBED BELOW
/*******
/** APPLID1 XXXXXXXX VTAM APPLID OF ACTIVE IMS SYSTEM
/** APPLID2 XXXXXXXX VTAM APPLID OF XRF ALTERNATE SYSTEM
/** APPLID3 XXXXXXXX VTAM APPLID OF RSR TRACKING SYSTEM
/** RES      X     BLOCK RESIDENT (N = NO, Y = YES)
/** FRE      XXXXX NUMBER OF FORMAT REQUEST ELEMENTS
/** PST      XXX   NUMBER OF PST'S PERMANENTLY ALLOC
/** MAXPST   XXX   MAXIMUM NUMBER OF PST'S
/** SAV      XXX   NUMBER OF DYNAMIC SAVE AREA SETS
/** SRCH     X     IMODULE LOAD LIBRARY SEARCH INDICATOR
/**                0 = SEARCH JPA AND LIBS BEFORE LPA
/**                1 = SEARCH JPA AND LPA BEFORE LIBS
/** SOD      X     SPIN OFF DUMP SYSOUT CLASS
/** VAUT     X     VTAM AUTH PATH OPTION (1=YES,0=NO)
/** FMTO          T = ONLINE FORMATTED DUMP WITH
/**                STORAGE IMAGE DELETIONS.
/**                OFFLINE SDUMPS PERMITTED FOR
/**                NON-IMS TERMINATING ERRORS.
/**                P = FULL ONLINE FORMATTED DUMP.
/**                OFFLINE SDUMPS PERMITTED FOR
/**                NON-IMS TERMINATING ERRORS.
/**                F = FULL ONLINE FORMATTED DUMP.
/**                OFFLINE SDUMPS SUPPRESSED FOR
/**                NON-IMS TERMINATING ERRORS.
/**                N = NO FORMATTED DUMP, NO OFFLINE
/**                DUMP. OFFLINE SDUMPS PERMITTED
/**                FOR NON-IMS TERMINATING ERRORS
/**                Z = NO FORMATTED DUMP, NO OFFLINE
/**                DUMP. OFFLINE SDUMPS
/**                SUPPRESSED FOR NON-IMS
/**                TERMINATING ERRORS.
/** (DEFAULT)   D = OFFLINE DUMP, OR ONLINE FOR-
/**                MATTED DUMP WITH STORAGE IMAGE
/**                DELETIONS IF OFFLINE DUMPING
/**                FAILS. OFFLINE SDUMPS
/**                PERMITTED FOR NON-IMS
/**                TERMINATING ERRORS.
/**                X = OFFLINE DUMP, OR ONLINE FOR-
/**                MATTED DUMP WITH STORAGE IMAGE
/**                DELETIONS IF OFFLINE DUMPING
/**                FAILS. OFFLINE SDUMPS
/**                SUPPRESSED FOR NON-IMS
/**                TERMINATING ERRORS.
/**                M = OFFLINE DUMP, ONLINE IMS DUMP
/**                FORMATTING NOT PERMITTED.
```



```

/**          OFFLINE SDUMPS PERMITTED FOR
/**          NON-IMS TERMINATING ERRORS.
/**          R = OFFLINE DUMP, ONLINE IMS DUMP
/**          FORMATTING NOT PERMITTED.
/**          OFFLINE SDUMPS SUPPRESSED FOR
/**          NON-IMS TERMINATING ERRORS.
/** AUTO    X    AUTOMATIC RESTART OPTION
/**          Y = AUTOMATIC RESTART DESIRED
/**          N = NO AUTOMATIC RESTART
/** IMSID   XXXX  IMS DCCTL SUBSYSTEM ID
/** NLXB    XXX   NUMBER OF PARALLEL VTAM SESSIONS
/** APPC    X     Y = ACTIVATE APPC/IMS
/**          N = DO NOT ACTIVATE APPC/IMS
/** LTERM   X     Y = LTERM USED IN DFSAPPC PROCESS
/**          N = LTERM NOT USED IN DFSAPPC
/**          PROCESS
/** ARMRST  X     Y = ALLOW MVS ARM TO RESTART
/**          N = ARM NOT RESTART IMS
/** DBRCNM  XXXXXXXX DBRC PROCLIB MEMBER NAME
/** PRDR    XXXXXXXX IMSRDR PROCLIB MEMBER NAME
/** SSM     XXXX   EXT SUBSYSTEM PROCLIB MEMBER ID
/** WADS    X     SINGLE OR DUAL WRITE AHEAD OPTION
/**          S=SINGLE, D=DUAL
/** ARC     XX    AUTOMATIC ARCHIVING OPTION
/**          0 = NOT AUTOMATIC
/**          1-99 = AUTOMATIC
/** RECA    XXX   RECEIVE ANY BUFFERS
/** FESTIM  XXXX  FRONTENDSWITCH TIMEOUT
/** RECASZ  XXXXX  RECEIVE ANY BUFFER SIZE
/** CRC     X     COMMAND RECOGNITION CHARACTER
/** TSR     X     U = UTC TIME
/**          L = LOCAL TIME (DEFAULT)
/** YEAR4   X     N = 2-DIGIT DATE
/**          Y = 4-DIGIT DATE
/** DC      XXX   DC PROC MEMBER SUFFIX IN
/**          IMS.PROCLIB
/**          DEFAULT VALUE IS 000
/** OTMA    X     Y = OTMA ENABLED
/**          N = OTMA NOT ENABLED
/**          DEFAULT VALUE IS N
/** GRNAME  XXXXXXXX OTMA XCF GROUP NAME
/**          NO DEFAULT VALUE
/** GRSNAME XXXXXXXX GENERIC RESOURCE GROUP
/**          NAME
/**          NO DEFAULT VALUE
/**
/** ***** SECURITY PARAMETERS *****
/**
/** AOIS    X     ICMD SECURITY OPTION
/** APPCSE  X     C = APPC RACF SECURITY IS CHECK
/**          F = APPC RACF SECURITY IS FULL
/**          N = APPC RACF SECURITY IS NONE
/**          P = APPC RACF SECURITY IS PROFILE
/** CMDMCS  X     MCS/EMCS COMMAND OPTION
/**          N=COMMANDS NOT ALLOWED WITH CRC
/**          Y=ALL COMMANDS ALLOWED WITH CRC
/**          R=RACF COMMAND SECURITY
/**          C=DFSCCMD0 COMMAND SECURITY
/**          B=RACF AND DFSCCMD0 CMD SEC
/** ISIS    X     RESOURCE ACCESS SECURITY OPTION
/**          0 = NO RESOURCE ACCESS SECURITY

```

DCC Procedure

```
/**          1 = RACF RESOURCE ACCESS SECURITY
/**          2 = USER RESOURCE ACCESS SECURITY
/**   RCF      X   RACF TRAN/CMD AUTHORIZATION
/**          N = NO TRAN/CMD AUTHORIZATION
/**          C = ETO CMD AUTHORIZATION
/**          T = TRAN AUTHORIZATION
/**          S = STATIC LTERM CMD AUTH
/**          Y = C AND T
/**          A = C AND T AND S
/**   RVFY     X   RACF REVERIFY OPTION
/**          Y = YES, N = NO
/**   SGN      X   SIGNON AUTHORIZATION CHECKING
/**          F = MTO CANNOT NEGATE LOADING
/**          SIGNON VERIFICATION SECURITY
/**          TABLE(SVST) FROM IMS.MATRIX
/**          V = SVST LOADED UNLESS OVERRIDEN
/**          BY MTO
/**          N = SVST NOT LOADED UNLESS
/**          OVERRIDEN BY MTO
/**          M = SINGLE USERID CAN SIGNON
/**          TO MULTIPLE STATIC TERMINALS
/**          G = 'F' + 'M'
/**          Z = 'Y' + 'M'
/**   TRN      X   TRANSACTION AUTHORIZATION CHECKING
/**          F = FORCED, Y = YES, N = NO
/**   RCFTCB   XX  NUMBER OF RCF TCB'S
/**
/******* MESSAGE QUEUE PARAMETERS *****
/**
/**   EXVR     X   PAGEFIX QMGR BUFFER POOLS
/**          (1=YES, 0=NO)
/**   QBUF     XXXX NUMBER OF MESSAGE QUEUE BUFFERS
/**   QTU      XXX  MSG QUEUE UPPER THRESHOLD (%)
/**   QTL      XXX  MSG QUEUE LOWER THRESHOLD (%)
/**
/******* SHARED QUEUES PARAMETERS *****
/**
/**   LGMSGSZ XXXXX LONG MESSAGE SIZE
/**   QBUFMAX XXXX  MAX NUMBER OF MSG QUEUE BUFFERS
/**   QBUFSZ   XXXXX SIZE OF MESSAGE QUEUE BUFFERS
/**   SHMSGSZ XXXXX SHORT MESSAGE SIZE
/**   SHAREDQ  XXX  SQ PROC MEMBER SUFFIX IN
/**          IMS.PROCLIB
/**          NO DEFAULT VALUE
/**
/******* ETO PARAMETERS *****
/**
/**   ETO      X   Y = EXTENDED TERMINAL OPTION
/**          N = NO EXTENDED TERMINAL OPTION
/**          M = NO EXTENDED TERMINAL OPTION
/**          BUT LOGON USERDATA SUPPORTED
/**          FOR STATIC TERMINALS
/**   ASOT     XXXX ETO AUTO SIGNOFF TIME
/**   ALOT     XXXX ETO AUTO LOGNOFF TIME
/**   DLQT     XXX  ETO DEAD LETTER QUEUE SIZE
/**
/******* RSR PARAMETERS *****
/**
/**   RSRMBR   XX   SUFFIX FOR RSR MEMBER
/**   TRACK    XXX  NO = NO RECOVERY TRACKING DONE
/**          RLT = RECOVERY TRACKING DONE
```

```

/** USERVAR XXXXXXXX USER NAME OF ACTIVE IMS SYSTEM FOR RSR
/**
/******* HASH TABLE PARAMETERS *****
/**
/** CHTS   XXXXX # OF CCB HASH TABLE SLOTS
/** LHTS   XXXXX # OF CNT HASH TABLE SLOTS
/** NHTS   XXXXX # OF VTCB HASH TABLE SLOTS
/** UHTS   XXXXX # OF SPQB HASH TABLE SLOTS
/**
/******* FAST PATH PARAMETERS *****
/**
/** EPCB   XXXX  EPCB POOL SIZE (1K BLOCKS)
/** EMHL   XXXXX SIZE OF EMH BUFFER IN BYTES
/**
/******* STORAGE POOL VALUES IN K, M OR G *****
/**
/** FBP    XXXXXX MESSAGE FORMAT BLOCK POOL SIZE
/** PSB    XXXXXX DCCTL PSB POOL SIZE
/** CIOP   XXXXXX TP DEVICE I/O POOL UPPER LIMIT
/** WKAP   XXXXXX WORKING STORAGE POOL SIZE
/** PSBW   XXXXXX PSB WORK POOL SIZE
/** HIOP   XXXXXX HIGH I/O POOL UPPER LIMIT
/** EMHB   XXXXXX EMHB POOL UPPER LIMIT
/** LUMP   XXXXXX LUMP POOL UPPER LIMIT
/** LUMC   XXXXXX LUMC POOL UPPER LIMIT
/** AOIP   XXXXXX AOI POOL UPPER LIMIT
/**
/******* MEMBER SUFFIXES *****
/**
/** SUF    X     CONTROL PGM NAME (DCCTL NUCLEUS)
/** FIX    XX    PAGE FIX NAME (DFSFIX..)
/** PRLD   XX    MODULE PRELOAD NAME (DFSAMPL..)
/** VSPEC  XX    TRACE/LOG DEF (DFSVM..)
/** SPM    XX    STORAGE POOL OPTIONS (DFSSPM..)
/** DSCT   X     ETO USER DESCRIPTOR TABLE(DFSDSCTX)
/**
/******* XRF PARAMETERS *****
/**
/** HSBID  X     XRF SYSTEM IDENTIFICATION
/**                1 FOR FIRST SYSTEM
/**                2 FOR SECOND SYSTEM
/** HSBMBR XX    XRF OPTIONS (DFSHSB..)
/**
/*******
/**                XRF SYSTEM DATA SET INFORMATION                *
/**                *
/**                *
/*******
/** THE FOLLOWING DATA SETS MUST RESIDE ON SHARED
/** VOLUMES:
/**
/** DDNAMES:  DFSOLPXX, DFSOLSXX
/**                DFSWADSX
/**                IMSRDS   - FIRST RDS
/**                IMSRDS2  - SECOND RDS (MANDATORY)
/**                MODSTAT  - FIRST MODSTAT
/**                MODSTAT2 - 2ND MODSTAT (MANDATORY)
/**                RECON1, RECON2, RECON3
/**
/** THE FOLLOWING DATA SETS MUST BE REPLICATED AND

```

DCC Procedure

```
/** SHOULD BE ON NON-SHARED VOLUMES. IF YOUR XRF
/** CONFIGURATION REQUIRES THAT BOTH IMS SUBSYSTEMS
/** BE EXECUTABLE ON EITHER CEC, THEN THESE DATA
/** SETS MUST BE ON SHARED VOLUMES.
/**
/**      DDNAMES:   IMSSPA
/**                  LGMSG, LGMSG
/**                  QBLKS, QBLKSL
/**                  SHMSG, SHMSG
/**
/**                  SYSABEND, SYSUDUMP
/**                  IMSMON
/**
/** THE FOLLOWING DATA SETS MAY BE REPLICATED. IF
/** REPLICATED, THESE DATA SETS SHOULD BE ON
/** NON-SHARED VOLUMES. IF THESE DATA SETS ARE NOT
/** REPLICATED, THEY MUST BE ON SHARED VOLUMES. IF
/** YOUR XRF CONFIGURATION REQUIRES THAT BOTH IMS
/** SUBSYSTEMS BE EXECUTABLE ON EITHER CEC,
/** THESE DATA SETS MUST BE ON SHARED VOLUMES.
/**
/**      DDNAMES:   FORMATA, FORMATB
/**                  IMSACBA, IMSACBB
/**                  IMSTFMTA, IMSTFMTB
/**                  MATRIXA, MATRIXB
/**                  MODBLKSA, MODBLKSB
/**                  PROCLIB, RESLIB
/**                  OTHER STEPLIB DATA SETS
/**
/** THE FOLLOWING DATA SETS SHOULD ALSO BE CONSIDERED
/** FOR REPLICATION ON EITHER SHARED OR NON-SHARED
/** VOLUMES (AS APPROPRIATE).
/**
/**      DATA SET:  JOBS   USED IN THE IMSRDR PROCEDURE
/**                  PGMLIB
/**                  PSBLIB USED BY GSAM
/**                  DBDLIB USED BY GSAM
/**
/** THE FOLLOWING IMS DATA SETS MAY ALSO BE IMPACTED
/** BY THE CONFIGURATION OF YOUR XRF COMPLEX.
/**
/** INSTALIB - USED DURING INSTALLATION/IVP
/** RESLIB  - CREATED BY SYSDEF - MANAGED BY SMP
/** PROCLIB - CREATED BY SYSDEF
/** MACLIB  - CREATED BY SYSDEF - MANAGED BY SMP
/** OPTIONS - CREATED BY SYSDEF - USED BY SMP
/** OBJDSET - CREATED BY SYSDEF
/** MODBLKS - CREATED BY SYSDEF - MANAGED BY SMP
/** MATRIX  - ONLINE CHANGE STAGING LIBRARY
/** ACBLIB  - ONLINE CHANGE STAGING LIBRARY
/** FORMAT  - ONLINE CHANGE STAGING LIBRARY
/** TFORMAT - ONLINE CHANGE STAGING LIBRARY
/** REFERAL - USED IN CONJUNCTION WITH FORMAT
/** DBSOURCE - USED BY SYSDEF - MANAGED BY SMP
/** DCSOURCE - USED BY SYSDEF - MANAGED BY SMP
/** SRSOURCE - USED BY SYSDEF - MANAGED BY SMP
/** GENLIB  - USED BY SYSDEF - MANAGED BY SMP
/** GENLIBA - USED BY SYSDEF - MANAGED BY SMP
/** GENLIBB - USED BY SYSDEF - MANAGED BY SMP
/** LOAD    - USED BY SYSDEF - MANAGED BY SMP
/** JCLLIB  - MANAGED BY SMP
```

```

/** SMP DATA SETS
/**
/*******
/**
/**          DATA DEFINITION STATEMENTS FOLLOW          *
/*******
/**
/******* LIBRARY STATEMENTS *****
/**
/**STEPLIB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
/**PROCLIB DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR
/**
/*******
/**
/** IN ORDER TO START A DEPENDENT REGION, MODIFIED
/** START-UP JCL IS WRITTEN FROM INTERNAL STORAGE TO
/** THE INTERNAL READER.
/**
/**IMSIRD DD SYSOUT=(A,INTRDR)
/**
/******* DASD LOGGING STATEMENTS *****
/**
/** THE FOLLOWING DD CARDS DESCRIBE THE DASD LOGGING
/** OLDS AND WADS. THESE CARDS ARE FOR EXAMPLE ONLY.
/** ALL OLDS AND WADS DATA SETS MAY BE DYNAMICALLY
/** ALLOCATED. DD CARDS ARE NOT REQUIRED.
/** THE OLDS AND WADS TO BE USED DURING STARTUP MUST
/** BE SPECIFIED VIA OLDSDEF AND WADSDEF CONTROL
/** STATEMENTS IN THE DFSVSMXX MEMBER OF IMS PROCLIB.
/** THE ACTUAL SELECTION OF OLDS AND WADS MUST BE
/** TAILORED TO INSTALLATION REQUIREMENTS. THE OLDS
/** AND WADS MUST BE PREDEFINED BY A SET UP JOB.
/** THE BLOCK SIZE OF ALL OLDS MUST BE THE SAME.
/** THE BLOCK SIZE AND DEVICE TYPE OF ALL WADS MUST
/** BE THE SAME. AT LEAST 3 PRIMARY OLDS AND 1 WADS
/** MUST BE AVAILABLE FOR STARTUP. THE BLOCK SIZE
/** SHOULD NOT BE SPECIFIED IN THIS JCL. THE LOGGER
/** WILL GET THE BLOCK SIZE FROM THE VTOC.
/**
/**DFSOLP00 DD DSN=IMS.&SYS.OLP00,DISP=SHR
/**DFSOLP01 DD DSN=IMS.&SYS.OLP01,DISP=SHR
/**DFSOLP02 DD DSN=IMS.&SYS.OLP02,DISP=SHR
/**DFSOLP03 DD DSN=IMS.&SYS.OLP03,DISP=SHR
/**DFSOLP04 DD DSN=IMS.&SYS.OLP04,DISP=SHR
/**DFSOLP05 DD DSN=IMS.&SYS.OLP05,DISP=SHR
/**
/**DFSOLS00 DD DSN=IMS.&SYS.OLS00,DISP=SHR
/**DFSOLS01 DD DSN=IMS.&SYS.OLS01,DISP=SHR
/**DFSOLS02 DD DSN=IMS.&SYS.OLS02,DISP=SHR
/**DFSOLS03 DD DSN=IMS.&SYS.OLS03,DISP=SHR
/**DFSOLS04 DD DSN=IMS.&SYS.OLS04,DISP=SHR
/**DFSOLS05 DD DSN=IMS.&SYS.OLS05,DISP=SHR
/**
/**DFSWADS0 DD DSN=IMS.&SYS.WADS0,DISP=SHR
/**DFSWADS1 DD DSN=IMS.&SYS.WADS1,DISP=SHR
/**
/******* MONITOR LOGGING STATEMENTS *****
/**
/** THE IMSMON DD STATEMENT MUST BE REMOVED IF
/** THIS DATA SET IS TO BE DYNAMICALLY ALLOCATED.

```

DCC Procedure

```
/**
//IMSMON DD DSN=IMS.&SYS1.IMSMON,DISP=(,KEEP),
// VOL=(,.,99),UNIT=(&LOGT,.,DEFER)
/**
//***** MESSAGE QUEUE STATEMENTS *****
/**
//QBLKS DD DSN=IMS.&SYS1.QBLKS,DISP=OLD
//SHMSG DD DSN=IMS.&SYS1.SHMSG,DISP=OLD
//LGMSG DD DSN=IMS.&SYS1.LGMSG,DISP=OLD
//QBLKSL DD DSN=IMS.&SYS1.QBLKSL,DISP=OLD
//SHMSGSL DD DSN=IMS.&SYS1.SHMSGSL,DISP=OLD
//LGMSGSL DD DSN=IMS.&SYS1.LGMSGSL,DISP=OLD
/**
//***** ONLINE CHANGE STATEMENTS *****
/**
//IMSACBA DD DSN=IMS.&SYS2.ACBLIBA,DISP=SHR
//IMSACBB DD DSN=IMS.&SYS2.ACBLIBB,DISP=SHR
//MODBLKSA DD DSN=IMS.&SYS2.MODBLKSA,DISP=SHR
//MODBLKSB DD DSN=IMS.&SYS2.MODBLKSB,DISP=SHR
//MODSTAT DD DSN=IMS.&SYS.MODSTAT,DISP=SHR
//MODSTAT2 DD DSN=IMS.&SYS.MODSTAT2,DISP=SHR
/**
//***** MFS STATEMENTS *****
/**
//FORMATA DD DSN=IMS.&SYS2.FORMAT,DISP=SHR
//FORMATB DD DSN=IMS.&SYS2.FORMATB,DISP=SHR
//IMSTFMTA DD DSN=IMS.&SYS2.TFORMAT,DISP=SHR
// DD DSN=IMS.&SYS2.FORMAT,DISP=SHR
//IMSTFMTB DD DSN=IMS.&SYS2.TFORMAT,DISP=SHR
// DD DSN=IMS.&SYS2.FORMATB,DISP=SHR
/**
//***** DCCTL SYSTEM STATEMENTS *****
/**
//IMSRDS DD DSN=IMS.&SYS.RDS,DISP=SHR
//IMSRDS2 DD DSN=IMS.&SYS.RDS2,DISP=SHR
//MATRIXA DD DSN=IMS.&SYS2.MATRIXA,DISP=SHR
//MATRIXB DD DSN=IMS.&SYS2.MATRIXB,DISP=SHR
//*DFSTCF DD DSN=IMS.TCFSLIB,DISP=SHR
//PRINTDD DD SYSOUT=&SOUT
//SYSUDUMP DD SYSOUT=&SOUT,
// DCB=(LRECL=125,RECFM=FBA,BLKSIZE=3129),
// SPACE=(6050,300,.,ROUND)
/**
//***** TELEPROCESSING LINE STATEMENTS *****
//***** GENERATED FROM DCCTL DEFINITION *****
/**
//DD2740S DD UNIT=/00B2 *** IMS LINE 2
//DD2740M2 DD UNIT=/00B3 *** IMS LINE 3
//DD3270R DD UNIT=/00C9 *** IMS LINE 4
//DD3270L DD UNIT=/0430 *** IMS LINE 5 TERMINAL 1
// DD UNIT=/0432 *** IMS LINE 5 TERMINAL 2
//TIIFQLIB DD UNIT=/00B1 *** IMS LINE 6
//DD2741SW DD UNIT=/002C *** IMS LINE 7
//LCLPTR1 DD UNIT=/0B01 *** IMS LINE 9
//LCLPTR2 DD UNIT=/0B02 *** IMS LINE 10
//LCLPTR3 DD UNIT=/0B03 *** IMS LINE 11
//LCLPTR4 DD UNIT=/0B04 *** IMS LINE 12
//LCLRDR1 DD UNIT=/0A01 *** IMS LINE 13
//LCLRDR2 DD UNIT=/0A02 *** IMS LINE 14
//LCLRDR3 DD UNIT=/0A03 *** IMS LINE 15
//LCLRDR4 DD UNIT=/0A04 *** IMS LINE 16
```

```

//LCLTAPE DD VOL=(PRIVATE,,99),UNIT=/0C01 IMS LINE 17
//DDJTRL DD UNIT=/04C1 *** IMS LINE 18
//SPOOL1 DD DISP=SHR,DSN=IMS.&SYS1.SYS01
//SPOOL2 DD DISP=SHR,DSN=IMS.&SYS1.SYS02
//SPOOL3 DD DISP=SHR,DSN=IMS.&SYS1.SYS03
//SPOOL4 DD DISP=SHR,DSN=IMS.&SYS1.SYS04
//SPOOL5 DD DISP=SHR,DSN=IMS.&SYS1.SYS05
//SPOOL6 DD DISP=SHR,DSN=IMS.&SYS1.SYS06
//LCLDISK1 DD SYSOUT=&SOUT *** UNIT=/0230 IMS LINE 21
//LCLDISK2 DD SYSOUT=&SOUT *** UNIT=/0231 IMS LINE 22
//SPOOL7 DD DISP=SHR,DSN=IMS.&SYS1.SYS07
//SPOOL8 DD DISP=SHR,DSN=IMS.&SYS1.SYS08
//SPOOL9 DD DISP=SHR,DSN=IMS.&SYS1.SYS09
//SPOOL0 DD DISP=SHR,DSN=IMS.&SYS1.SYS010
//DD2741 DD UNIT=/00B4 *** IMS LINE 24
//DTWX3335 DD UNIT=/00D1 *** IMS LINE 25
//LOG2740 DD UNIT=/04C5 *** IMS LINE 26
//DDTTRL DD UNIT=/04C2 *** IMS LINE 27
//DDVTRL DD UNIT=/04C3 *** IMS LINE 28
//DDWTRL DD UNIT=/04C4 *** IMS LINE 29
//DDNDS01R DD UNIT=/00C5 *** IMS LINE 30
//DDNDS01L DD UNIT=/0320 *** IMS LINE 31 TERMINAL 1
// DD UNIT=/0321 *** IMS LINE 31 TERMINAL 2
// DD UNIT=/0322 *** IMS LINE 31 TERMINAL 3
// DD UNIT=/0323 *** IMS LINE 31 TERMINAL 4
// DD UNIT=/0324 *** IMS LINE 31 TERMINAL 5
//DDNDS0L2 DD UNIT=/0325 *** IMS LINE 32 TERMINAL 1
// DD UNIT=/0326 *** IMS LINE 32 TERMINAL 2
//DDAD201R DD UNIT=/012E *** IMS LINE 33
//DDAD202R DD UNIT=/012F *** IMS LINE 34
//DDAD201L DD UNIT=/0301 *** IMS LINE 35 TERMINAL 1
// DD UNIT=/0302 *** IMS LINE 35 TERMINAL 2
// DD UNIT=/0303 *** IMS LINE 35 TERMINAL 3
// DD UNIT=/0304 *** IMS LINE 35 TERMINAL 4
// DD UNIT=/0305 *** IMS LINE 35 TERMINAL 5
// DD UNIT=/0306 *** IMS LINE 35 TERMINAL 6
//DDAD20L2 DD UNIT=/030E *** IMS LINE 36 TERMINAL 1
// DD UNIT=/030F *** IMS LINE 36 TERMINAL 2
//D3270FP1 DD UNIT=/0533 *** IMS LINE 37 TERMINAL 1
//D3270FP2 DD UNIT=/0534 *** IMS LINE 38 TERMINAL 1
//D3270FP3 DD UNIT=/0535 *** IMS LINE 39 TERMINAL 1
//D3270FP4 DD UNIT=/0536 *** IMS LINE 40 TERMINAL 1
//D3270FP5 DD UNIT=/0537 *** IMS LINE 41 TERMINAL 1
//D3270FP6 DD UNIT=/0538 *** IMS LINE 42 TERMINAL 1
//DD3270RF DD UNIT=/01C9 *** IMS LINE 43
//DD3270S DD UNIT=/01C8 *** IMS LINE 44
//MSC23ASS DD UNIT=/00CA
//MSC2CASS DD UNIT=/00BA
//MSC23UNA DD UNIT=/00CB
//*
//***** EXTERNAL SUBSYSTEM STATEMENTS *****
//*
//* INCLUDE THE DFSESL DD STATEMENT IF AN EXTERNAL
//* SUBSYSTEM CONNECTION TO DB2 IS DESIRED.
//*

```

DCC Parameters

The following parameters are valid for the DCC procedure. See “Parameter Descriptions” on page 244.

DCC Parameters

ALOT=	NHTS=
AOIP=	NLXB=
AOIS=	OTMA=
APPC=	OTMNM=
APPCSE=	OTMASE=
APPLID1=	PRDR=
APPLID2=	PRLD=
APPLID3=	PSB=
ARC=	PSBW=
ARMRST=	PST=
ASOT=	QBUF=
AUTO=	QBUFMAX=
CHTS=	QBUFSZ=
CIOP=	QTL=
CMDMCS=	QTU=
CRC=	RCF=
DBRCNM=	RECA=
DC=	RECASZ=
DLQT=	RES=
DPRTY=	RGN=
DSCT=	RGSUF=
EMHB=	RSRMBR=
EMHL=	RVFY=
EPCB=	SAV=
ETO=	SGN=
EXVR=	SHAREQ=
FBP=	SHMSGSZ=
FESTIM=	SOD=
FIX=	SOUT=
FMTO=	SPM=
FRE=	SRCH=
GRNAME=	SSM=
GRSNAME=	SUF=
HIOP=	SYS=
HSBID=	SYS1=
HSMBR=	SYS2=
IMSID=	TRACK=
ISIS=	TRN=
LGMSGSZ=	TSR=
LHTS=	UHTS=
LOGT=	USERVAR=
LTERM=	VAUT=
LUMC=	VSPEC=
LUMP=	WADS=
MAXPST=	WKAP=
	YEAR4=

DCC DD Statements

The following DD statements are valid for the DCC procedure. See “DD Statement Descriptions” on page 279.

DFSOLPnn DD	MODSBLKSB DD
DFSOLSnn DD	MODSTAT DD
DFSWADSn DD	MODSTAT2 DD

FORMATA DD	PRINT DD
FORMATB DD	PROCLIB DD
IMSACBA DD	QBLKS DD
IMSACBB DD	QBLKSL DD
IMSMON DD	RECON1 DD
IMSRDS DD	RECON2 DD
IMSRDS2 DD	RECON3 DD
IMSFMTA DD	RESLIB DD
IMSFMTB DD	SHMSG DD
LGMSG DD	SHMSGL DD
LGMSGL DD	STEPLIB DD
MATRIXA DD	SYSABEND DD
MATRIXB DD	SYSUDUMP DD
MODBLKSA	

DFSMPR Procedure

Procedure Description

The following procedure executes an IMS message processing address space.

```
//      PROC SOUT=A,RGN=56K,SYS2=,
//      CL1=001,CL2=000,CL3=000,CL4=000,
//      OPT=N,OVLA=0,SPIE=0,VALCK=0,TLIM=00,
//      PCB=000,PRLD=,STIMER=,SOD=,DBLDL=,
//      NBA=,OBA=,IMSID=,AGN=,VSFX=,VFREE=,
//      SSM=,PREINIT=,ALTID=,PWFI=N,
//      APARAM=,LOCKMAX=,APPLFE=
//*
//REGION EXEC PGM=DFSRR00,REGION=&RGN,
//      TIME=1440,DPRTY=(12,0),
//      PARM=(MSG,&CL1&CL2&CL3&CL4,
//      &OPT&OVLA&SPIE&VALCK&TLIM&PCB,
//      &PRLD,&STIMER,&SOD,&DBLDL,&NBA,
//      &OBA,&IMSID,&AGN,&VSFX,&VFREE,
//      &SSM,&PREINIT,&ALTID,&PWFI,
//      '&APARM',&LOCKMAX),&APPLFE
//*
//STEPLIB DD DSN=IMS.&SYS2.PGMLIB,DISP=SHR
//      DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
//PROCLIB DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR
//SYSUDUMP DD SYSOUT=&SOUT,
//      DCB=(LRECL=121,BLKSIZE=3129,RECFM=VBA),
//      SPACE=(125,(2500,100),RLSE,,ROUND)
```

DFSMPR Parameters

The following parameters are valid for the DFSMPR procedure. See "Parameter Descriptions" on page 244.

AGN=	PREINIT=
ALTID=	PRLD=
APARM=	PWFI=
APPLFE=	RGN=
CL1=,CL2=,CL3=,CL4=	SOD=
DBLDL=	SOUT=
IMSID=	SPIE=
LOCKMAX=	SSM=
MSG=	STIMER=
NBA=	SYS2=

DFSMPR Procedure

OBA=
OPT=
OVLA=
PCB=

TLIM=
VALCK=
VFREE=
VSFX=

DFSMPR DD Statements

The following DD statements are valid for the DFSMPR procedure. See “DD Statement Descriptions” on page 279.

DFSESL DD¹
FPTRACE DD¹
PROCLIB DD

STEPLIB DD
SYSUDUMP DD

Note: ¹. These DD statements are optional.

DLIBATCH Procedure

To run DL/I batch jobs in an RSR environment, you need to be aware of the following:

- The batch-only databases that are identified to be tracked must be included in the system definition (DATABASE macro) for the tracking subsystem.
- The default GSG name and TMI name used by batch jobs are specified in the IMSCTRL macro, but they can be overridden in the DLIBATCH procedure.
- Batch jobs do not use the DFSRSRxx PROCLIB member, and are therefore limited to the default VTAM modename (TMDEFLT).

Procedure Description

The following is a one-step procedure for an offline DL/I batch processing program using PSB and DBD libraries.

If VSAM databases are used, refer to “Specifying IMS Buffer Pools” on page 378.

```
//      PROC MBR=TEMPNAME,PSB=,BUF=7,  
//      SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,  
//      SRCH=0,CKPTID=,MON=N,LOGA=0,FMTO=T,  
//      IMSID=,SWAP=,DBRC=,IRLM=,IRLMNM=,  
//      BKO=N,IQB=,SSM=,APARM=,  
//      RGN=2048K,  
//      SOUT=A,LOGT=2400,SYS2=,  
//      LOCKMAX=,GSGNAME=,TMINAME=  
//G     EXEC PGM=DFSRR00,REGION=&RGN,  
//      PARM=(DLI,&MBR,&PSB,&BUF,  
//      &SPIE&TEST&EXCPVR&RST,&PRLD,  
//      &SRCH,&CKPTID,&MON,&LOGA,&FMTO,  
//      &IMSID,&SWAP,&DBRC,&IRLM,&IRLMNM,  
//      &BKO,&IQB,&SSM,'&APARM',  
//      &LOCKMAX,&GSGNAME,&TMINAME)  
//STEPLIB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR  
//      DD DSN=IMS.&SYS2.PGMLIB,DISP=SHR  
//DFSRESLB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR  
//IMS     DD DSN=IMS.&SYS2.PSBLIB,DISP=SHR  
//      DD DSN=IMS.&SYS2.DBDLIB,DISP=SHR  
//PROCLIB DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR  
//IEFRDRR DD DSN=IMSLOG,DISP=(,KEEP),VOL=(,.,99),  
//      UNIT=(&LOGT,,DEFER),  
//      DCB=(RECFM=VB,BLKSIZE=1920,  
//      LRECL=1916,BUFNO=2)
```

```
//IEFRDER2 DD DSN=IMSLOG2,DISP=(,KEEP),VOL=(.,.,99),
//          UNIT=(&LOGT,,DEFER,SEP=IEFRDER),
//          DCB=(RECFM=VB,BLKSIZE=1920,
//          LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&SOUT,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=605),
//          SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
```

Notes:

- Parameters in parentheses are positional.
- The IEFRDER statement is not required if the job does not declare database-update intent.

For a job step declaring database-update intent, DD DUMMY can be specified if the job step is not using DBRC. This is valid where an image copy of the database is taken before the update job step.

Log initialization calculates the smallest value necessary for logical record length. If the JCL logical record length value is larger than the calculated value, the JCL value is used; otherwise, log initialization uses the calculated value for logical record length and adds 4 for the block size.

If multiple volumes are required for the system log, a volume count value should be specified in the VOL parameter of the DD statement.

If the IBM 3480 tape drive is used for the IMS log data set, IMS forces tape write mode (DCB=OPCD=W). The default on the 3480 is to serve as a buffer for the write so that IMS cannot detect when the write is actually performed. If a power failure occurs after a log record is written to the 3480, and the database is updated but the log record is not yet written to tape, database integrity is lost. Tape write mode is forced for the log in batch and for GSAM data sets.

DLIBATCH Parameters

The following parameters are valid for the DLIBATCH procedure. See “Parameter Descriptions” on page 244.

APARM=	MBR=
BKO=	MON=
BUF=	PRLD=
CKPTID=	PSB=
DBRC=	RGN=
EXCPVR=	RST=
FMTO=	SOUT=
GSGNAME=	SPIE=
IMSID=	SRCH=
IOB=	SSM=
IRLM=	SWAP=
IRLMNM=	SYS2=
LOCKMAX=	TEST=
LOGA=	TMINAME=
LOGT=	

DLIBATCH DD Statements

The following DD statements are valid for the DLIBATCH procedure. See “DD Statement Descriptions” on page 279.

DFSRESLB DD

IMSLOGR DD

DLIBATCH Procedure

DFSSTAT DD¹
DFSVSAMP DD
IEFRDER DD¹
IEFRDER2 DD¹
IMS DD

IMSMON DD
PROCLIB DD
STEPLIB DD
SYSABEND DD
SYSUDUMP DD

Note:¹These statements are optional.

DLISAS Procedure

Procedure Description

This procedure initializes a DL/I separate address space (DLISAS).

Related Reading: For information on LSO=S see “DL/I Address Space” in the *IMS/ESA Administration Guide: System*.

The following parameters in the IMS procedure are applicable with the LSO=S option: DLINM, CSAPSB, and DLIPSB.

DLINM specifies the partitioned data set (PDS) member name that IMS uses for the DLISAS procedure. The following list shows how the member name is assigned.

1. DLINM=DLISAS default name
2. DLINM=member in SYSGEN IMSCTRL macro; overrides DLISAS default name
3. DLINM=member defined in DFSPBxxx, (xxx specifies the specific IMS.PROCLIB member); overrides SYSGEN value
4. DLINM=member in JCL EXEC parameter; overrides the IMS.PROCLIB member value

As specified by the DLINM parameter on the IMSCTRL macro, the cataloged procedure (for the DLISAS) is stored in the IMS.PROCLIB member during system definition. This member must be copied to SYS1.PROCLIB.

CSAPSB and DLIPSB specify the PSB pool sizes and override the values specified with the SASPSB parameter on the BUFPOOLS system definition macro.

Default process: An internally defined MVS START command is issued for the DLINM value and is coded to override the PARM=(DLS,&IMSID) procedure statement. The override defines the IMSID of the subsystem issuing the START, allowing a generic procedure to be used by multiple IMS control regions.

IMS does not issue the IMS READY message until the DLISAS successfully initiates. If the started DLISAS procedure is in error, you should correct and restart it.

When the DL/I address space connects to the IMS control region, the DLISAS procedure name is compared with the DLINM value. If they differ, the DLISAS abends.

Alternate process: You can supply a START command (in member=DLINM of a PDS defined in the control region JCL //PROCLIB) instead of using the internally defined MVS START command. Columns 1-5 must be “START”, or message DFS1930I is issued and the default process is performed.

Here are two types of START commands that you can specify:

- START IMSRDR, MBR=DLISAS: the specified PDS member is read in as a job (rather than as a started task) through the MVS internal reader.
- START DLIAA, PARM=(DLS, IMSA): DL/I executes as a started task and DLIAA connects to IMSA.

For active and inactive ACBLIBs, the data set names and their concatenation order must be identical (same DSN and VOLSER) in the DLISAS procedure and in the IMS procedure.

IMS.PROCLIB must be defined in the DLISAS procedure and in the IMS procedure. Sometimes the same DLINM member is read from IMS.PROCLIB by both the control region and DLISAS.

RACF and Resource Protection

If the resources (DL/I databases, for example) are RACF-protected, the user ID associated with DLISAS procedure must be authorized to access them.

Related Reading: For more information on how to authorize the user ID, see *IMS/ESA Administration Guide: System*.

The DLISAS procedure follows:

```
//          PROC RGN=3072K,DPTY='(14,15)',SOUT=A,
//          IMSID=SYS3,SYS2=
//IEFPROC EXEC PGM=DFSMVRC0,REGION=&RGN,
//          DPTY=&DPTY,PARM=(DLS,&IMSID)
//*****
//*
//STEPLIB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
//PROCLIB DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR
//***** ACBLIB *****
//*
//* THE SPECIFICATION OF THE ACBLIB DATASETS
//* IN THE DLI/SAS REGION PROCEDURE MUST
//* CORRESPOND EXACTLY WITH THE SPECIFICATION
//* IN THE CONTROL REGION JCL
//*
//IMSACBA DD DSN=IMS.&SYS2.ACBLIBA,DISP=SHR
//IMSACBB DD DSN=IMS.&SYS2.ACBLIBB,DISP=SHR
//SYSUDUMP DD SYSOUT=&SOUT.
//SYSABEND DD SYSOUT=&SOUT.
//***** DATA BASE DD CARDS *****
//*
//* USER MAY OPTIONALLY SUPPLY THE DD STATEMENTS
//* FOR THE ON-LINE DATA BASES TO BE
//* INSERTED HERE PRIOR TO ATTEMPTING
//* AN ON-LINE SYSTEM EXECUTION USING
//* THIS PROCEDURE.
//* IF NO DD STATEMENTS ARE SUPPLIED FOR
//* A DATA BASE, IMS ASSUMES THAT THIS
//* DATA BASE HAS BEEN DESCRIBED THROUGH
//* THE DFSMDA MACRO.
//*
```

DLISAS Parameters

The following parameters are valid for the DLISAS procedure. See “Parameter Descriptions” on page 244.

DPTY=
IMSID=
RGN=

SOUT=
SYS2=

DLISAS Procedure

DLISAS DD Statements

In addition to the following DD statements, statements for DL/I databases must be in this procedure, and not in the IMS procedure; add statements for data sets representing databases that are not to be dynamically allocated. DD Statements for Fast Path databases remain in the IMS procedure. See "DD Statement Descriptions" on page 279.

IMSACBA DD
IMSACBB DD
PROCLIB DD

STEPLIB DD
SYSUDUMP DD

DXRJPROC Procedure

Procedure Description

The following procedure starts the Internal Resource Lock Manager (IRLM). This procedure is supplied on the IRLM distribution tape.

Related Reading: For information on obtaining the IRLM from the distribution tape and preparing it for use, refer to *IMS/ESA Installation Volume 1: Installation and Verification*.

When installing the IRLM, a unique copy of DXRJPROC should be created for each IRLM that might execute concurrently. The procedure name and parameters should be changed, as required, for each IRLM. Because no default value is provided for the IRLMID parameter, the DXRJPROC procedure must at least be altered to assign a value for IRLMID.

In a configuration in which two systems are sharing data, the two IRLMs must execute concurrently. A typical way to define this configuration is:

- Procedure name IRLM1 with IRLMID=1
- Procedure name IRLM2 with IRLMID=2

For testing purposes, two IRLMs can communicate with one another using XCF while executing concurrently on a single system. A typical way to define this environment is:

- Procedure name JRLM1 with IRLMID=1 and IRLMNM=JRLM
- Procedure name KRLM8 with IRLMID=8 and IRLMNM=KRLM

IRLM has a global deadlock manager. This is designated as the IRLM with the lowest IRLMID. IRLMs in a parallel sysplex group dynamically readjust the global manager identity as members join and leave the group. If you are concerned about the placement of the global deadlock manager, you can put the lowest IRLMID on a specific processor.

Startup Procedure for IRLM 2.1

The following procedure executes IRLM 2.1.

```
//DXRJPROC PROC RGN=3072K,  
//          IRLMNM=IRLM,  
//          IRLMID=,  
//          SCOPE=LOCAL,  
//          DEADLOK='5,1',  
//          MAXCSA=8,  
//          PC=NO,
```

```
//          MAXUSRS=,
//          IRLMGRP=
//          LOCKTAB=IRLMT1,
//          TRACE=YES
//          EXEC PGM=DXRRLM00,DPRTY=(15,15),
//          PARM=(&IRLMNM,&IRLMID,&SCOPE,&DEADLOK,&MAXCSA;
//          &PC,&MAXUSRS,&IRLMGRP,&LOCKTABL,&TRACE),
//          REGION=&RGN.
//STEPLIB DD DSN=x,DISP=SHR
//SYSABEND DD SYSOUT=A
```

DXRJPROC Parameter Descriptions for IRLM 2.1

The following parameters are valid for the DXRJPROC when you use IRLM 2.1. See “Parameter Descriptions” on page 244.

DEADLOK=	MAXUSRS=
IRLMGRP=	MAXCSA=
IRLMID=	PC=
IRLMNM=	SCOPE=
LOCKTAB=	TRACE=

DXRJPROC DD Statements for IRLM 2.1

The following DD statements are valid for the DXRJPROC when you are using IRLM 2.1. See “DD Statement Descriptions” on page 279.

STEPLIB DD	SYSABEND DD
-------------------	--------------------

FDR Procedure

Procedure Description

The following FDR procedure executes a Fast Database Recovery (FDBR) address space:

```
//          PROC RGN=2000K,SOUT=A,DPTY='(14,15)',
//          SYS=,SYS2=,
//          RGSUF=IMS,PARM1=,PARM2=
//IEFPROC EXEC PGM=DFSMVRCO,DPRTY=&DPTY,
//          REGION=&RGN,
//          PARM='FDR,&RGSUF,&PARM1,&PARM2'
//*
//*
//* THE MEANING AND MAXIMUM SIZE OF EACH PARAMETER
//* IS AS FOLLOWS:
//*
//***** CONTROL REGION SPECIFICATIONS *****
//*****
//* RGSUF XXX EXEC PARM DEFAULT BLOCK SUFFIX FOR
//* MEMBER DFSPBXXX.
//*****
//*
//* PARM1 , PARM2 PARAMETERS BOTH ARE USED TO SPECIFY
//* CHARACTER STRINGS THAT CONTAIN IMS KEYWORD
//* PARAMETERS. I.E. PARM1='AUTO=Y,PST=222,RES=Y'
//*
//* ALL OF THE VALID IMS KEYWORD PARAMETERS
//* ARE DESCRIBED BELOW
//*****
```

FDR Procedure

```
/** FMTO      T = ONLINE FORMATTED DUMP WITH
/**          STORAGE IMAGE DELETIONS.
/**          OFFLINE SDUMPS PERMITTED FOR
/**          NON-IMS TERMINATING ERRORS.
/**          P = FULL ONLINE FORMATTED DUMP.
/**          OFFLINE SDUMPS PERMITTED FOR
/**          NON-IMS TERMINATING ERRORS.
/**          F = FULL ONLINE FORMATTED DUMP.
/**          OFFLINE SDUMPS SUPPRESSED FOR
/**          NON-IMS TERMINATING ERRORS.
/**          N = NO FORMATTED DUMP, NO OFFLINE
/**          DUMP. OFFLINE SDUMPS PERMITTED
/**          FOR NON-IMS TERMINATING ERRORS
/**          Z = NO FORMATTED DUMP, NO OFFLINE
/**          DUMP. OFFLINE SDUMPS
/**          SUPPRESSED FOR NON-IMS
/**          TERMINATING ERRORS.
/** (DEFAULT) D = OFFLINE DUMP, OR ONLINE FORMAT-
/**          TED DUMP WITH STORAGE IMAGE
/**          DELETIONS IF OFFLINE DUMPING
/**          FAILS. OFFLINE SDUMPS
/**          PERMITTED FOR NON-IMS
/**          TERMINATING ERRORS.
/**          X = OFFLINE DUMP, OR ONLINE FORMAT-
/**          TED DUMP WITH STORAGE IMAGE
/**          DELETIONS IF OFFLINE DUMPING
/**          FAILS. OFFLINE SDUMPS
/**          SUPPRESSED FOR NON-IMS
/**          TERMINATING ERRORS.
/**          M = OFFLINE DUMP, ONLINE IMS DUMP
/**          FORMATTING NOT PERMITTED.
/**          OFFLINE SDUMPS PERMITTED FOR
/**          NON-IMS TERMINATING ERRORS.
/**          R = OFFLINE DUMP, ONLINE IMS DUMP
/**          FORMATTING NOT PERMITTED.
/**          OFFLINE SDUMPS SUPPRESSED FOR
/**          NON-IMS TERMINATING ERRORS.
/** IMSID XXXX IMS SUBSYSTEM IDENTIFIER
/** ARMST X    Y = ALLOW MVS ARM TO RESTART
/**          N = ARM NOT RESTART IMS
/** IRLNM XXXX IRLM SUBSYSTEM NAME
/** WADS X     SINGLE OR DUAL WADS,S=SINGLE,D=DUAL
/** UHASH XXXXXXXX USER HASH MODULE NAME
/** GRNAME XXXXXXXX OTMA XCF GROUP NAME
/**          NO DEFAULT VALUE
/**
/******* FDR PARAMETER *****
/**
/** FDRMBR XX  SUFFIX FOR FDR MEMBER IN
/**          IMS.PROCLIB
/******* FAST PATH PARAMETERS *****
/**
/** BSIZ XXXXX DATA BASE BUFFER SIZE
/** DBBF XXXXX NUMBER OF DATABASE BUFFERS
/** LGNR XX   NUMBER OF LOG ENTRIES IN DEDB BUFFERHEADER
/******* STORAGE POOL VALUES IN K, M OR G *****
/**
/** PSB XXXXXX PSB POOL SIZE - NON DLISAS
/** DMB XXXXXX DMB POOL SIZE
/** WKAP XXXXXX WORKING STORAGE BUFFER POOL SIZE
/** PSBW XXXXXX PSB WORK POOL SIZE
```



```

/** DBWP XXXXXX DATABASE WORK POOL SIZE
/** CSAPSB XXXXXX DLISAS: CSA PSB POOL SIZE
/** DLIPSB XXXXXX DLISAS: DLI PSB POOL SIZE
/**
/******* MEMBER SUFFIXES *****
/**
/**  SUF    X      LAST CHARACTER OF CTL PROGRAM LOAD
/**                MODULE MEMBER NAME
/**  VSPEC  XX     2 CHARACTER BUFFER POOL SPEC MODULE SUFFIX
/**  SPM    XX     STG POOL MGR PROCLIB MEMBER SUFFIX
/**
/*******
/**
/**STEPLIB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
/**PROCLIB DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR
/**
/******* DASD LOGGING DD CARDS *****
/** THE FOLLOWING DD CARDS DESCRIBE THE DASD LOGGING
/** OLDS AND WADS. THESE CARDS ARE FOR EXAMPLE ONLY.
/** ALL OLDS AND WADS DATA SETS MAY BE DYNAMICALLY
/** ALLOCATED. DD CARDS ARE NOT REQUIRED.
/** THE OLDS AND WADS TO BE USED DURING STARTUP MUST
/** BE SPECIFIED VIA OLDSDEF AND WADSDEF CONTROL
/** STATEMENTS IN THE DFSVSMXX MEMBER OF IMS PROCLIB.
/** THE ACTUAL SELECTION OF OLDS AND WADS MUST BE
/** TAILORED TO INSTALLATION REQUIREMENTS. THE OLDS
/** AND WADS MUST BE PREDEFINED BY A SET UP JOB.
/** THE BLOCK SIZE OF ALL OLDS MUST BE THE SAME.
/** THE BLOCK SIZE AND DEVICE TYPE OF ALL WADS MUST
/** BE THE SAME. AT LEAST 3 PRIMARY OLDS AND 1 WADS
/** MUST BE AVAILABLE FOR STARTUP. THE BLOCK SIZE
/** SHOULD NOT BE SPECIFIED IN THIS JCL. THE LOGGER
/** WILL GET THE BLOCK SIZE FROM THE VTOC.
/**
/**DFSOLP00 DD DSN=IMS.&SYS.OLP00,DISP=SHR
/**DFSOLP01 DD DSN=IMS.&SYS.OLP01,DISP=SHR
/**DFSOLP02 DD DSN=IMS.&SYS.OLP02,DISP=SHR
/**DFSOLP03 DD DSN=IMS.&SYS.OLP03,DISP=SHR
/**DFSOLP04 DD DSN=IMS.&SYS.OLP04,DISP=SHR
/**DFSOLP05 DD DSN=IMS.&SYS.OLP05,DISP=SHR
/**
/**DFSOLS00 DD DSN=IMS.&SYS.OLS00,DISP=SHR
/**DFSOLS01 DD DSN=IMS.&SYS.OLS01,DISP=SHR
/**DFSOLS02 DD DSN=IMS.&SYS.OLS02,DISP=SHR
/**DFSOLS03 DD DSN=IMS.&SYS.OLS03,DISP=SHR
/**DFSOLS04 DD DSN=IMS.&SYS.OLS04,DISP=SHR
/**DFSOLS05 DD DSN=IMS.&SYS.OLS05,DISP=SHR
/**
/**DFSWADS0 DD DSN=IMS.&SYS.WADS0,DISP=SHR
/**DFSWADS1 DD DSN=IMS.&SYS.WADS1,DISP=SHR
/**
/**IMSACBA DD DSN=IMS.&SYS2.ACBLIBA,DISP=SHR
/**IMSACBB DD DSN=IMS.&SYS2.ACBLIBB,DISP=SHR
/**MODBLKSA DD DSN=IMS.&SYS2.MODBLKSA,DISP=SHR
/**MODBLKSB DD DSN=IMS.&SYS2.MODBLKSB,DISP=SHR
/**MODSTAT DD DSN=IMS.&SYS.MODSTAT,DISP=SHR
/******* SYSTEM STATEMENTS *****
/**
/**SYSUDUMP DD SYSOUT=&SOUT,
/**          DCB=(LRECL=125,RECFM=FBA,BLKSIZE=3129),
/**          SPACE=(6050,300,,ROUND)

```

FDR Procedure

```
//IMSRDS DD DSN=IMS.&SYS.RDS,DISP=SHR
//*
//***** DATA BASE DD CARDS *****
//*
//* USER MAY OPTIONALLY SUPPLY THE DD STATEMENTS
//* FOR THE ON-LINE DATA BASES TO BE
//* INSERTED HERE PRIOR TO ATTEMPTING
//* AN ON-LINE SYSTEM EXECUTION USING
//* THIS PROCEDURE.
//* IF NO DD STATEMENTS ARE SUPPLIED FOR
//* A DATA BASE, IMS ASSUMES THAT THIS
//* DATA BASE HAS BEEN DESCRIBED THROUGH
//* THE DFSMDA MACRO.
//* IF THE USER WILL BE EXECUTING WITH THE DL/I
//* SAS OPTION, THESE DD STATEMENTS SHOULD BE ADDED
//* TO THE DLISAS PROCLIB MEMBER OR DESCRIBED
//* THROUGH THE DFSMDA MACRO.
//*
//***** DBRC RECON DD CARDS *****
//*
//* USER MAY OPTIONALLY SUPPLY THE DD CARDS
//* REQUIRED FOR THE DBRC RECON DATA SET.
//* IF NO DD STATEMENTS ARE SUPPLIED FOR RECON
//* DATASETS, IMS ASSUMES THAT THE DATASETS
//* HAVE BEEN DESCRIBED THROUGH THE DFSMDA MACRO.
//*
```

FDR Procedure Parameters

The following parameters are valid for the FDR procedure. See “Parameter Descriptions” on page 244.

ARMRST=	PSB=
BSIZ=	PSBW=
CSAPSB= ¹	RGN=
DBBF=	RGSUF=
DBWP=	SOUT=
DLIPSB= ¹	SPM=
DMB=	SUF=
DPRTY=,	SYS=
FDRMBR=	SYS2=
FMTO=	UHASH=
GRNAME=	VSPEC=
IMSID=	WADS=
IRLMNM=	WKAP=
LGNR=	

¹ For the FDR procedure, the sum of the values for the CSAPSB and DLIPSB defines the PSB pool size. If PSB is also specified, the larger value (PSB or the sum of CSAPSB and DLIPSB) is used.

FDR Procedure DD Statements

The following DD statements are valid for the FDR procedure. In addition to the following DD statements, add statements for data sets that are not to be dynamically allocated. See “DD Statement Descriptions” on page 279.

DFSOLPn DD

JCLPDS DD

DFSOLSn DD	MODSBLKSA DD
DFSWADSn DD	MODBLKSB DD
IMSACBA DD	MODSTAT DD
IMSACBB DD	PROCLIB DD
IMSRDS DD	STEPLIB DD
JCLOUT DD	SYSUDUMP DD

FPUTIL Procedure

Procedure Description

The following procedure executes the Fast Path utility programs with Data Entry Database (DEDB) online.

```
//FPUTIL PROC SOUT=A,RGN=128K,SYS2=,
//          DBD=,REST=00,DIRCA=002,
//          PRLD=,IMSID=,AGN=,SSM=,ALTID=
//FPU EXEC PGM=DFSRR00,REGION=&RGN,
//          PARM=(IFP,&DBD,DBF#FPU0,&REST,00,,1,
//          &DIRCA,&PRLD,0,,, &IMSID,&AGN,&SSM,,
//          &ALTID)
//STEPLIB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
//PROCLIB DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR
//SYSPRINT DD SYSOUT=&SOUT.
//SYSUDUMP DD SYSOUT=&SOUT,
//          DCB=(LRECL=121,RECFM=VBA,BLKSIZE=3129),
//          SPACE=(125,(2500,100),RLSE,,ROUND)
```

Note: Parameters in parentheses are positional.

NBA= and OBA= are not valid parameters for this procedure. The following default values are used: NBA=7 and OBA=0.

FPUTIL Procedure Parameters

The following parameters are valid for the FPUTIL procedure. See "Parameter Descriptions" on page 244.

AGN=	REST=
ALTID=	RGN=
DBD=	SOUT=
DIRCA=	SSM=
IMSID=	SYS2=
PRLD=	

IMS Procedure

Procedure Description

The following procedure executes a DB/DC online control program.

RACF and Resource Protection: If the resources (for example, DL/I databases) are RACF protected, the user ID associated with the IMS procedure must be authorized to access them.

Related Reading: For more information on how to authorize the user ID, see *IMS/ESA Administration Guide: System*.

IMS Procedure

DLISAS Procedure: If you specify LSO=S in the IMS procedure, thereby automatically invoking the DLISAS procedure, the DD statements for the DL/I databases must be in the DLISAS procedure, and not in the IMS procedure. Dynamic allocation members remain in the STEPLIB library.

Program Specification Block (PSB) Pools: If you do not specify LSO=S in the IMS procedure, one PSB pool is created, as specified by the PSB parameter on the BUFPOOLS macro. If you do specify LSO=S, two PSB pools are created and the PSB parameter is ignored. The pool sizes are specified with the SASPSB parameter on the BUFPOOLS system definition macro, and overridden with the CSAPSB and DLIPSB parameters on the IMS procedure.

Related Reading: For an overview of information about the DL/I address space options, see *IMS/ESA Administration Guide: System*.

The control region procedure contains the following parameters:

```
IMSPROC &RGN=, &SOUT=, &DPTY=, &LOGT=, &RGSUF=000,
        &PARM1=, &PARM2=
```

The PARM2 parameter is needed because MVS:

- Allows up to 100 characters in the EXEC parameter area
- Does not allow symbolic parameters to be continued on next record
- Does not allow more than 66 characters for each keyword option phrase on the START command.

The following is a sample of the IMS procedure.

```
//      PROC RGN=2000K,SOUT=A,DPTY='(14,15)',
//          SYS=,SYS1=,SYS2=,
//          LOGT=2400,
//          RGSUF=IMS,PARM1=,PARM2=
//IEFPROC EXEC PGM=DFSMVRC0,DPRTY=&DPTY,
//          REGION=&RGN,
//          PARM='CTL,&RGSUF,&PARM1,&PARM2'
//*
//*
//* THE MEANING AND MAXIMUM SIZE OF EACH PARAMETER
//* IS AS FOLLOWS:
//*
//***** CONTROL REGION SPECIFICATIONS *****
//*****
//* RGSUF   XXX   EXEC PARM DEFAULT BLOCK SUFFIX FOR
//*                               MEMBER DFSPBXXX.
//*****
//*
//* PARM1 , PARM2 PARAMETERS BOTH ARE USED TO SPECIFY
//* CHARACTER STRINGS THAT CONTAIN IMS KEYWORD
//* PARAMETERS. I.E. PARM1='AUTO=Y,PST=222,RES=Y'
//*
//* ALL OF THE VALID IMS KEYWORD PARAMETERS
//* ARE DESCRIBED BELOW
//*****
//* APPLID1 XXXXXXXX VTAM APPLID OF ACTIVE IMS SYSTEM
//* APPLID2 XXXXXXXX VTAM APPLID OF XRF ALTERNATE SYSTEM
//* APPLID3 XXXXXXXX VTAM APPLID OF RSR TRACKING SYSTEM
//* RES     X     BLOCK RESIDENT (N = NO, Y = YES)
//* FRE     XXXXX NUMBER OF FORMAT REQUEST ELEMENTS
//* PST     XXX   NUMBER OF PST'S PERMANENTLY ALLOC
//* MAXPST  XXX   MAXIMUM NUMBER OF PST'S
```

```

/** SAV    XXX  NUMBER OF DYNAMIC SAVE AREA SETS
/** SRCH   X    MODULE SEARCH INDICATOR FOR DIRECTED LOAD
/**
/**        0 = STANDARD SEARCH
/**        1 = SEARCH JPA AND LPA BEFORE PDS
/** SOD    X    1 CHARACTER SYSOUT CLASS
/** VAUT   X    VTAM AUTH PATH OPTION (1=YES,0=NO)
/** FMTO
/**        T = ONLINE FORMATTED DUMP WITH
/**          STORAGE IMAGE DELETIONS.
/**          OFFLINE SDUMPS PERMITTED FOR
/**          NON-IMS TERMINATING ERRORS.
/**        P = FULL ONLINE FORMATTED DUMP.
/**          OFFLINE SDUMPS PERMITTED FOR
/**          NON-IMS TERMINATING ERRORS.
/**        F = FULL ONLINE FORMATTED DUMP.
/**          OFFLINE SDUMPS SUPPRESSED FOR
/**          NON-IMS TERMINATING ERRORS.
/**        N = NO FORMATTED DUMP, NO OFFLINE
/**          DUMP. OFFLINE SDUMPS PERMITTED
/**          FOR NON-IMS TERMINATING ERRORS
/**        Z = NO FORMATTED DUMP, NO OFFLINE
/**          DUMP. OFFLINE SDUMPS
/**          SUPPRESSED FOR NON-IMS
/**          TERMINATING ERRORS.
/** (DEFAULT) D = OFFLINE DUMP, OR ONLINE FOR-
/**          MATTED DUMP WITH STORAGE IMAGE
/**          DELETIONS IF OFFLINE DUMPING
/**          FAILS. OFFLINE SDUMPS
/**          PERMITTED FOR NON-IMS
/**          TERMINATING ERRORS.
/**        X = OFFLINE DUMP, OR ONLINE FOR-
/**          MATTED DUMP WITH STORAGE IMAGE
/**          DELETIONS IF OFFLINE DUMPING
/**          FAILS. OFFLINE SDUMPS
/**          SUPPRESSED FOR NON-IMS
/**          TERMINATING ERRORS.
/**        M = OFFLINE DUMP, ONLINE IMS DUMP
/**          FORMATTING NOT PERMITTED.
/**          OFFLINE SDUMPS PERMITTED FOR
/**          NON-IMS TERMINATING ERRORS.
/**        R = OFFLINE DUMP, ONLINE IMS DUMP
/**          FORMATTING NOT PERMITTED.
/**          OFFLINE SDUMPS SUPPRESSED FOR
/**          NON-IMS TERMINATING ERRORS.
/** AUTO   X    Y = AUTOMATIC RESTART DESIRED
/**        N = NO AUTOMATIC RESTART
/** IMSID  XXXX  IMS SUBSYSTEM IDENTIFIER
/** NLXB   XXX   # ADD'L LXBS FOR MSC VTAM
/** LSO    X     Y = DL/I LOCAL STORAGE OPTION ON
/**        S = DLI/SAS OPTION
/**
/**
/** APPC   X    Y = ACTIVATE APPC/IMS
/**        N = DO NOT ACTIVATE APPC/IMS
/** LTERM  X    Y = LTERM USED IN DFSAPPC PROCESS
/**        N = LTERM NOT USED IN DFSAPPC
/**        PROCESS
/** ARMRST X    Y = ALLOW MVS ARM TO RESTART
/**        N = ARM NOT RESTART IMS
/** IRLM   X    Y = YES, N = NO
/** IRLMNM XXXX  IRLM SUBSYSTEM NAME
/** SSM    XXXX  EXT SUBSYSTEM PROCLIB MEMBER ID

```

IMS Procedure

```

/** WADS X SINGLE/DUAL WADS,S=SINGLE,D=DUAL
/** ARC XX AUTOMATIC ARCHIVE.
/** 0 = NOT AUTOMATIC
/** 1-99 = AUTOMATIC
/** UHASH XXXXXXXX USER HASH MODULE NAME
/** DBRCNM XXXXXXXX DBRC PROCLIB MEMBER NAME
/** DLINM XXXXXXXX DL/I PROCLIB MEMBER NAME
/** PRDR XXXXXXXX IMSRDR PROCLIB MEMBER NAME
/** FESTIM XXXX FRONTENDSWITCH TIMEOUT (SECONDS)
/** RECASZ XXXXX RECEIVE ANY BUFFER SIZE
/** PIMAX XXXXXX ENQ/DEQ POOL MAXIMUM BYTES
/** PIINCR XXXXXX ENQ/DEQ POOL INCREMENT
/** RECA XXX NUMBER OF RECEIVE ANY BUFFERS
/** CRC X COMMAND RECOGNITION CHARACTER
/** TSR X U = UTC TIME
/** L = LOCAL TIME (DEFAULT)
/** YEAR4 X N = 2-DIGIT DATE
/** Y = 4-DIGIT DATE
/** DC XXX DC PROC MEMBER SUFFIX IN
/** IMS.PROCLIB
/** DEFAULT VALUE IS 000
/** OTMA X Y = OTMA ENABLED
/** N = OTMA NOT ENABLED
/** DEFAULT VALUE IS N
/** GRNAME XXXXXXXX OTMA XCF GROUP NAME
/** NO DEFAULT VALUE
/** GRSNAME XXXXXXXX GENERIC RESOURCE GROUP
/** NAME
/** NO DEFAULT VALUE
/** IMSGROUP XXXX IMS GROUP NAME
/** NO DEFAULT VALUE
/**
/******* SECURITY PARAMETERS *****
/**
/** AOIS X ICMD SECURITY OPTION
/** APPCSE X C = APPC RACF SECURITY IS CHECK
/** F = APPC RACF SECURITY IS FULL
/** N = APPC RACF SECURITY IS NONE
/** P = APPC RACF SECURITY IS PROFILE
/** CMDMCS X MCS/EMCS COMMAND OPTION
/** N=COMMANDS NOT ALLOWED WITH CRC
/** Y=ALL COMMANDS ALLOWED WITH CRC
/** R=RACF COMMAND SECURITY
/** C=DFSCCMD0 COMMAND SECURITY
/** B=RACF AND DFSCCMD0 CMD SEC
/** ISIS X 0 = NO RESOURCE ACCESS SECURITY
/** 1 = RACF RESOURCE ACCESS SECURITY
/** 2 = USER RESOURCE ACCESS SECURITY
/** RCF X RACF USED FOR TRANS. AND SIGNON
/** A = Y + S, Y = T + C, S = S + C.
/** RVFY X RACF REVERIFY OPTION
/** Y = YES, N = NO
/** SGN X SIGNON AUTHORIZATION CHECKING
/** F = MTO CANNOT NEGATE LOADING
/** SIGNON VERIFICATION SECURITY
/** TABLE(SVST) FROM IMS.MATRIX
/** Y = SVST LOADED UNLESS OVERRIDEN
/** BY MTO
/** N = SVST NOT LOADED UNLESS
/** OVERRIDEN BY MTO
/** M = SINGLE USERID CAN SIGNON

```

```

/**          TO MULTIPLE STATIC TERMINALS
/**          G = 'F' + 'M'
/**          Z = 'Y' + 'M'
/** TRN    X    TRANSACTION AUTHORIZATION CHECKING
/**          F = FORCED, Y = YES, N = NO
/** RCFTCB XX  NUMBER OF RCF TCB'S
/**
/******* MESSAGE QUEUE PARAMETERS *****
/**
/** EXVR    X    PAGEFIX QMGR BUFFER POOLS
/**          (1=YES, 0=NO)
/** QBUF    XXXX NUMBER OF MESSAGE QUEUE BUFFERS
/** QTL     XXX  QUEUE LOWER THRESHOLD (%)
/** QTU     XXX  QUEUE UPPER THRESHOLD (%)
/**
/******* SHARED QUEUES PARAMETERS *****
/**
/** LGMSGSZ XXXXX LONG MESSAGE SIZE
/** QBUFMAX XXXX  MAX NUMBER OF MSG QUEUE BUFFERS
/** QBUFSZ  XXXXX SIZE OF MESSAGE QUEUE BUFFERS
/** SHMSGSZ XXXXX SHORT MESSAGE SIZE
/** SHAREDQ XXX   SQ PROC MEMBER SUFFIX IN
/**          IMS.PROCLIB
/**          NO DEFAULT VALUE
/**
/******* XRF PARAMETERS *****&sup1;
/**
/** HSBID   X    XRF SYSTEM ID
/**          1 FOR FIRST SYSTEM
/**          2 FOR SECOND SYSTEM
/**
/** HSBMBR  XX   SUFFIX FOR XRF MEMBER IN
/**          IMS.PROCLIB
/**          00 IS DEFAULT
/**
/******* FDR PARAMETER *****
/**
/** FDRMBR  XX   SUFFIX FOR FDR MEMBER IN
/**          IMS.PROCLIB
/**
/******* FAST PATH PARAMETERS *****
/**
/** BSIZ    XXXXX DATA BASE BUFFER SIZE
/** OTHR    XXX  NUMBER OF OUTPUT THREADS
/** DBFX    XXXXX SYSTEM ALLOCATION OF DATA BASE BUFFERS TO BE
/**          FIXED AT START OF 1ST FAST PATH DEP REGION
/** DBBF    XXXXX NUMBER OF DATABASE BUFFERS
/** MSDB    X    SUFFIX FOR MSDB MEMBER ON
/**          IMS.PROCLIB
/** LGNR    XX   NUMBER OF LOG ENTRIES IN DEDB BUFFERHEADER
/** EMHL    XXXXX SIZE OF EMH BUFFER IN BYTES
/**
/**
/******* ETO PARAMETERS *****
/**
/** ETO     X    Y = EXTENDED TERMINAL OPTION
/**          N = NO EXTENDED TERMINAL OPTION
/**          M = NO EXTENDED TERMINAL OPTION
/**          BUT LOGON USERDATA SUPPORTED
/**          FOR STATIC TERMINALS
/** ASOT    XXXX ETO AUTO SIGNOFF TIME

```

IMS Procedure

```
//* ALOT XXXX ETO AUTO LOGNOFF TIME
//* DLQT XXX ETO DEAD LETTER QUEUE SIZE
//*
//***** RSR PARAMETERS *****&sup2;
//*
//* RSRMBR XX SUFFIX FOR RSR MEMBER
//* TRACK XXX NO = NO RECOVERY TRACKING DONE
//* RLT = RECOVERY TRACKING DONE
//* DLT = DATABASE TRACKING DONE
//* USERVAR XXXXXXXX USER NAME OF ACTIVE IMS SYSTEM FOR RSR
//*
//***** HASH TABLE PARAMETERS *****
//*
//* CHTS XXXXX # OF CCB HASH TABLE SLOTS
//* LHTS XXXXX # OF CNT HASH TABLE SLOTS
//* NHTS XXXXX # OF VTCB HASH TABLE SLOTS
//* UHTS XXXXX # OF SPQB HASH TABLE SLOTS
//*
//***** STORAGE POOL VALUES IN K, M OR G *****
//*
//* FBP XXXXXX MESSAGE BUFFER POOL SIZE
//* PSB XXXXXX PSB POOL SIZE - NON DLISAS
//* DMB XXXXXX DMB POOL SIZE
//* CIOP XXXXXX CIOP POOL UPPER LIMIT
//* WKAP XXXXXX WORKING STORAGE BUFFER POOL SIZE
//* PSBW XXXXXX PSB WORK POOL SIZE
//* DBWP XXXXXX DATABASE WORK POOL SIZE
//* CSAPSB XXXXXX DLISAS: CSA PSB POOL SIZE
//* DLIPSB XXXXXX DLISAS: DLI PSB POOL SIZE
//* EPCB XXXXXX EPCB POOL SIZE
//* HIOP XXXXXX HIOP POOL UPPER LIMIT
//* FPWP XXXXXX FPWP POOL UPPER LIMIT
//* EMHB XXXXXX EMHB POOL UPPER LIMIT
//* LUMP XXXXXX LUMP POOL UPPER LIMIT
//* LUMC XXXXXX LUMC POOL UPPER LIMIT
//* AOIP XXXXXX AOI POOL UPPER LIMIT
//*
//***** MEMBER SUFFIXES *****
//*
//* SUF X LAST CHARACTER OF CTL PROGRAM LOAD
//* MODULE MEMBER NAME
//* FIX XX 2 CHARACTER FIX PROCEDURE MODULE SUFFIX
//* PRLD XX 2 CHARACTER PROCLIB MEMBER SUFFIX FOR PRELOAD
//* VSPEC XX 2 CHARACTER BUFFER POOL SPEC MODULE SUFFIX
//* SPM XX STG POOL MGR PROCLIB MEMBER SUFFIX
//* DSCT X ETO USER DESCRIPTOR TABLE(DFSDSCTX)
//*
//*****
//* *
//* DATA DEFINITION STATEMENTS FOLLOW *
//* *
//*****
//*
//***** LIBRARY STATEMENTS *****
//*
//STEPLIB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
//PROCLIB DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR
//*
//***** GENERIC START DEPENDANT REGION *****
//*
//* IN ORDER TO START A DEPENDENT REGION, MODIFIED
```



```

/** START-UP JCL IS WRITTEN FROM INTERNAL STORAGE TO
/** THE INTERNAL READER.
/**
/**IMSIRD DD SYSOUT=(A,INTRDR)
/**
/******* DASD LOGGING STATEMENTS *****
/**
/** THE FOLLOWING DD CARDS DESCRIBE THE DASD LOGGING
/** OLDS AND WADS. THESE CARDS ARE FOR EXAMPLE ONLY.
/** ALL OLDS AND WADS DATA SETS MAY BE DYNAMICALLY
/** ALLOCATED. DD CARDS ARE NOT REQUIRED.
/** THE OLDS AND WADS TO BE USED DURING STARTUP MUST
/** BE SPECIFIED VIA OLDSDEF AND WADSDEF CONTROL
/** STATEMENTS IN THE DFSVSMXX MEMBER OF IMS PROCLIB.
/** THE ACTUAL SELECTION OF OLDS AND WADS MUST BE
/** TAILORED TO INSTALLATION REQUIREMENTS. THE OLDS
/** AND WADS MUST BE PREDEFINED BY A SET UP JOB.
/** THE BLOCK SIZE OF ALL OLDS MUST BE THE SAME.
/** THE BLOCK SIZE AND DEVICE TYPE OF ALL WADS MUST
/** BE THE SAME. AT LEAST 3 PRIMARY OLDS AND 1 WADS
/** MUST BE AVAILABLE FOR STARTUP. THE BLOCK SIZE
/** SHOULD NOT BE SPECIFIED IN THIS JCL. THE LOGGER
/** WILL GET THE BLOCK SIZE FROM THE VTOC.
/**
/**DFSOLP00 DD DSN=IMS.&SYS.OLP00,DISP=SHR
/**DFSOLP01 DD DSN=IMS.&SYS.OLP01,DISP=SHR
/**DFSOLP02 DD DSN=IMS.&SYS.OLP02,DISP=SHR
/**DFSOLP03 DD DSN=IMS.&SYS.OLP03,DISP=SHR
/**DFSOLP04 DD DSN=IMS.&SYS.OLP04,DISP=SHR
/**DFSOLP05 DD DSN=IMS.&SYS.OLP05,DISP=SHR
/**
/**DFSOLS00 DD DSN=IMS.&SYS.OLS00,DISP=SHR
/**DFSOLS01 DD DSN=IMS.&SYS.OLS01,DISP=SHR
/**DFSOLS02 DD DSN=IMS.&SYS.OLS02,DISP=SHR
/**DFSOLS03 DD DSN=IMS.&SYS.OLS03,DISP=SHR
/**DFSOLS04 DD DSN=IMS.&SYS.OLS04,DISP=SHR
/**DFSOLS05 DD DSN=IMS.&SYS.OLS05,DISP=SHR
/**
/**DFSWADS0 DD DSN=IMS.&SYS.WADS0,DISP=SHR
/**DFSWADS1 DD DSN=IMS.&SYS.WADS1,DISP=SHR
/**
/******* MONITOR STATEMENTS *****
/**
/** THE IMSMON DD STATEMENT MUST BE REMOVED IF
/** THIS DATA SET IS TO BE DYNAMICALLY ALLOCATED.&sup3;
/**
/**IMSMON DD DSN=IMS.&SYS1.IMSMON,DISP=(,KEEP),
/** VOL=(,,99),UNIT=(&LOGT,,DEFER)&sup4;
/**
/******* MESSAGE QUEUE STATEMENTS *****
/**
/**QBLKS DD DSN=IMS.&SYS1.QBLKS,DISP=OLD
/**SHMSG DD DSN=IMS.&SYS1.SHMSG,DISP=OLD
/**LGMSG DD DSN=IMS.&SYS1.LGMSG,DISP=OLD
/**QBLKSL DD DSN=IMS.&SYS1.QBLKSL,DISP=OLD
/**SHMSGL DD DSN=IMS.&SYS1.SHMSGL,DISP=OLD
/**LGMSGL DD DSN=IMS.&SYS1.LGMSGL,DISP=OLD
/**
/******* ONLINE CHANGE STATEMENTS *****
/**
/**IMSACBA DD DSN=IMS.&SYS2.ACBLIBA,DISP=SHR

```

IMS Procedure

```
//IMSACBB DD DSN=IMS.&SYS2.ACBLIBB,DISP=SHR
//MODBLKSA DD DSN=IMS.&SYS2.MODBLKSA,DISP=SHR
//MODBLKSB DD DSN=IMS.&SYS2.MODBLKSB,DISP=SHR
//MODSTAT DD DSN=IMS.&SYS2.MODSTAT,DISP=SHR
//MODSTAT2 DD DSN=IMS.&SYS2.MODSTAT2,DISP=SHR&sup1;
//*
//***** FAST PATH STATEMENTS *****
//*
//MSDBCP1 DD DSN=IMS.&SYS2.MSDBCP1,DISP=SHR
//MSDBCP2 DD DSN=IMS.&SYS2.MSDBCP2,DISP=SHR
//MSDBCP3 DD DSN=IMS.&SYS2.MSDBCP3,DISP=SHR&sup1;
//MSDBCP4 DD DSN=IMS.&SYS2.MSDBCP4,DISP=SHR&sup1;
//MSDBDUMP DD DSN=IMS.&SYS1.MSDBDUMP,DISP=SHR
//MSDBINIT DD DSN=IMS.&SYS2.MSDBINIT,DISP=SHR
//*
//***** MFS STATEMENTS *****
//*
//FORMATA DD DSN=IMS.&SYS2.FORMAT,DISP=SHR
//FORMATB DD DSN=IMS.&SYS2.FORMATB,DISP=SHR
//IMSTFMTA DD DSN=IMS.&SYS2.TFORMAT,DISP=SHR
//IMSTFMTB DD DSN=IMS.&SYS2.TFORMAT,DISP=SHR
//IMSTFMTB DD DSN=IMS.&SYS2.TFORMAT,DISP=SHR
//IMSTFMTB DD DSN=IMS.&SYS2.FORMATB,DISP=SHR
//*
//***** SYSTEM STATEMENTS *****
//*
//SYSUDUMP DD SYSOUT=&SOUT,
//          DCB=(LRECL=125,RECFM=FBA,BLKSIZE=3129),
//          SPACE=(6050,300,,ROUND)
//IMSRDS DD DSN=IMS.&SYS2.RDS,DISP=SHR
//IMSRDS2 DD DSN=IMS.&SYS2.RDS2,DISP=SHR&sup1;
//MATRIXA DD DSN=IMS.&SYS2.MATRIXA,DISP=SHR
//MATRIXB DD DSN=IMS.&SYS2.MATRIXB,DISP=SHR
//*DFSTCF DD DSN=IMS.TCFSLIB,DISP=SHR
//PRINTDD DD SYSOUT=&SOUT
//*
//***** TELEPROCESSING LINE STATEMENTS *****
//***** GENERATED FROM THE DB/DC DEFINITION *****
//*
//DD2740S DD UNIT=/00B2 *** IMS LINE 2
//DD2740M2 DD UNIT=/00B3 *** IMS LINE 3
//DD3270R DD UNIT=/00C9 *** IMS LINE 4
//DD3270L DD UNIT=/0430 *** IMS LINE 5 TERMINAL 1
//          DD UNIT=/0432 *** IMS LINE 5 TERMINAL 2
//TIIFQLIB DD UNIT=/00B1 *** IMS LINE 6
//DD2741SW DD UNIT=/002C *** IMS LINE 7
//LCLPTR1 DD UNIT=/0B01 *** IMS LINE 9
//LCLPTR2 DD UNIT=/0B02 *** IMS LINE 10
//LCLPTR3 DD UNIT=/0B03 *** IMS LINE 11
//LCLPTR4 DD UNIT=/0B04 *** IMS LINE 12
//LCLRDR1 DD UNIT=/0A01 *** IMS LINE 13
//LCLRDR2 DD UNIT=/0A02 *** IMS LINE 14
//LCLRDR3 DD UNIT=/0A03 *** IMS LINE 15
//LCLRDR4 DD UNIT=/0A04 *** IMS LINE 16
//LCLTAPE DD VOL=(PRIVATE,,99),UNIT=/0C01 IMS LINE 17
//DDJTRL DD UNIT=/04C1 *** IMS LINE 18
//SPOOL1 DD DISP=SHR,DSN=IMS.&SYS1.SYS01
//SPOOL2 DD DISP=SHR,DSN=IMS.&SYS1.SYS02
//SPOOL3 DD DISP=SHR,DSN=IMS.&SYS1.SYS03
//SPOOL4 DD DISP=SHR,DSN=IMS.&SYS1.SYS04
//SPOOL5 DD DISP=SHR,DSN=IMS.&SYS1.SYS05
```

```

//SPOOL6 DD DISP=SHR,DSN=IMS.&SYS1.SYS06
//LCLDISK1 DD SYSOUT=&SOUT *** UNIT=/0230 IMS LINE 21
//LCLDISK2 DD SYSOUT=&SOUT *** UNIT=/0231 IMS LINE 22
//SPOOL7 DD DISP=SHR,DSN=IMS.&SYS1.SYS07
//SPOOL8 DD DISP=SHR,DSN=IMS.&SYS1.SYS08
//SPOOL9 DD DISP=SHR,DSN=IMS.&SYS1.SYS09
//SPOOL0 DD DISP=SHR,DSN=IMS.&SYS1.SYS010
//DD2741 DD UNIT=/00B4 *** IMS LINE 24
//DTWX3335 DD UNIT=/00D1 *** IMS LINE 25
//LOG2740 DD UNIT=/04C5 *** IMS LINE 26
//DDTTRL DD UNIT=/04C2 *** IMS LINE 27
//DDVTRL DD UNIT=/04C3 *** IMS LINE 28
//DDWTRL DD UNIT=/04C4 *** IMS LINE 29
//DDNDS01R DD UNIT=/00C5 *** IMS LINE 30
//DDNDS01L DD UNIT=/0320 *** IMS LINE 31 TERMINAL 1
// DD UNIT=/0321 *** IMS LINE 31 TERMINAL 2
// DD UNIT=/0322 *** IMS LINE 31 TERMINAL 3
// DD UNIT=/0323 *** IMS LINE 31 TERMINAL 4
// DD UNIT=/0324 *** IMS LINE 31 TERMINAL 5
//DDNDS0L2 DD UNIT=/0325 *** IMS LINE 32 TERMINAL 1
// DD UNIT=/0326 *** IMS LINE 32 TERMINAL 2
//DDAD201R DD UNIT=/012E *** IMS LINE 33
//DDAD202R DD UNIT=/012F *** IMS LINE 34
//DDAD201L DD UNIT=/0301 *** IMS LINE 35 TERMINAL 1
// DD UNIT=/0302 *** IMS LINE 35 TERMINAL 2
// DD UNIT=/0303 *** IMS LINE 35 TERMINAL 3
// DD UNIT=/0304 *** IMS LINE 35 TERMINAL 4
// DD UNIT=/0305 *** IMS LINE 35 TERMINAL 5
// DD UNIT=/0306 *** IMS LINE 35 TERMINAL 6
//DDAD20L2 DD UNIT=/030E *** IMS LINE 36 TERMINAL 1
// DD UNIT=/030F *** IMS LINE 36 TERMINAL 2
//D3270FP1 DD UNIT=/0533 *** IMS LINE 37 TERMINAL 1
//D3270FP2 DD UNIT=/0534 *** IMS LINE 38 TERMINAL 1
//D3270FP3 DD UNIT=/0535 *** IMS LINE 39 TERMINAL 1
//D3270FP4 DD UNIT=/0536 *** IMS LINE 40 TERMINAL 1
//D3270FP5 DD UNIT=/0537 *** IMS LINE 41 TERMINAL 1
//D3270FP6 DD UNIT=/0538 *** IMS LINE 42 TERMINAL 1
//DD3270RF DD UNIT=/01C9 *** IMS LINE 43
//DD3270S DD UNIT=/01C8 *** IMS LINE 44
//MSC23ASS DD UNIT=/00CA
//MSCTCASS DD UNIT=/00BA
//MSC23UNA DD UNIT=/00CB
//*
//***** EXTERNAL SUBSYSTEM STATEMENTS *****
//*
//* USER MAY OPTIONALLY ADD THE DFSESL DD CARD
//* FOR EXTERNAL SUBSYSTEM CONNECTION.
//*
//***** DATA BASE STATEMENTS *****&sup5;
//*
//* USER MAY OPTIONALLY SUPPLY THE DD STATEMENTS
//* FOR THE ON-LINE DATA BASES TO BE
//* INSERTED HERE PRIOR TO ATTEMPTING
//* AN ON-LINE SYSTEM EXECUTION USING
//* THIS PROCEDURE.
//* IF NO DD STATEMENTS ARE SUPPLIED FOR
//* A DATA BASE, IMS ASSUMES THAT THIS
//* DATA BASE HAS BEEN DESCRIBED THROUGH
//* THE DFSMDA MACRO.
//* IF THE USER WILL BE EXECUTING WITH THE DL/I
//* SAS OPTION, THESE DD STATEMENTS SHOULD BE ADDED

```

IMS Procedure

```
/** TO THE DLISAS PROCLIB MEMBER OR DESCRIBED  
/** THROUGH THE DFSMDA MACRO.  
/**
```

Notes:

1. XRF-capable system only.
2. RSR-capable system only.
3. Specify DCB=BLKSIZE=nnnnn to ensure good performance when running the IMS Monitor. A block size should be as large as possible for your environment. A value greater than 20,000 is usually optimal. If DCB=BLKSIZE is not specified, the default block size is 1048 for a JCL-allocated IMSMON dataset, and any dataset block size value (in the DSCB) is ignored. The default blocksize value is likely to cause performance degradation for a busy IMS system.
4. Produced only if requested in the IMSCTF macro.
5. DEDB areas must be allocated in the IMS control region address space, regardless of whether the DL/I SAS option is used. (DEDB dynamic allocation is generally done using information from the RECON data set.) This requires that one of the following is true:
 - The DD statements are in the control region JCL
 - The dynamic allocation members are in IMS.RESLIB, as addressed by JOBLIB and STEPLIB

If dynamic allocation is to be used, it is generally done using information from the DFSMDA members. If the databases are registered with DBRC and if DFSMDA is not available, dynamic allocation is done using information from the RECON data set.

It is recommended that you register the databases with DBRC and not use DFSMDA.

IMS Procedure Parameters

The following parameters are valid for the IMS procedure. See “Parameter Descriptions” on page 244.

ALOT=	LSO=
AOIP=	LTERM=
AOIS=	LUMC=
APPC=	LUMP=
APPCSE=	MAXPST=
APPLID1=name1	MSDB=¹
APPLID2=name2	NHTS=
APPLID3=name3	NLXB=
ARC=	OTMA=
	OTMAMD=
ARMRST=	OTMANM=
	OTMASE=
	OTMASP=
ASOT=	OTHR=¹
AUTO=	PIINCR=
BSIZ=¹	PIMAX=
CHTS=	PRLD=
CIOP=	PRDR=
CMDMCS	PSB=

CRC=	PSBW=
CSAPSB=	PST=
DBBF= ¹	QBUF=
DBRCNM=	QBUFMAX=
DBFX= ¹	QBUFSZ=
DBWP=	QTL=
DC=	QTU=
DLINM=	RCF=
DLIPSB=	RCFTCB=
DLQT=	RECA
DMB=	RECASZ=
DPRTY=	RES=
DSCT=	RGN=
EMHB=	RGSUF=
EMHL=	RSRMBR=
EPCB= ¹	RVFY=
ETO=	SAV=
EXVR=	SGN=
FBP=	SHAREDQ=
FDRMBR=	SHMSGSZ=
FESTIM=	SOD=
FIX=	SOUT=
FMTO=	SPM=
FPWP=	SRCH=
FRE=	SSM=
GRNAME=	SUF=
GRSNAME=	SYS=
HIOP=	SYS1=
HSBID=	SYS2=
HSBMBR=	TRACK=
MSGROUP=	TRN=
MSID=	TSR=
IRLM=	UHASH= ¹
IRLMNM=	UHTS=
ISIS=	USERVAR=
LGMSGSZ=	VAUT=
LGNR= ¹	VSPEC=
LHTS=	WADS=
LOGT=	WKAP=
LSO=	YEAR4=

Note: ¹Fast Path parameters

IMS Procedure DD Statements

The following parameters are valid for the IMS procedure. See “DD Statement Descriptions” on page 279.

In addition to the following DD statements, add statements for data sets representing IMS databases that are not to be dynamically allocated.

DFSDUPPM DD	MSDBCP2 DD
DFSESL DD ¹	MSDBCP3 DD
DFSOLPnn DD	MSDBCP 4 DD
DFSOLSnn DD	MSDBDUMP DD

IMS Procedure

DFSWADSn DD	MSDBINIT DD
FORMATA DD	MODBLKSA DD
FORMATB DD	MODBLKSB DD
IMSACBA DD	MODSTAT DD
IMSACBB DD	MODSTAT2 DD
IMSMON DD	PRINTDD DD
IMSRDS DD	PROCLIB DD
IMSRDS2 DD	QBLKS DD
IMSTFMTA DD	QBLKSL DD
IMSTFMTB DD	SHMSG DD
LGMSG DD	SHMSGL DD
LGMSGL DD	STEPLIB DD
MATRIXA DD	SYSUDUMP DD
MATRIXB DD	SYSTPRT DD
MSDBCP1 DD	

Note: ¹ Optional DD statement

IMSBATCH Procedure

The following procedure executes an DB/DC online batch message processing region.

```
//      PROC MBR=TEMPNAME,PSB=,IN=,OUT=,
//      OPT=N,SPIE=0,TEST=0,DIRCA=000,
//      PRLD=,STIMER=,CKPTID=,PARDLI=,
//      CPUTIME=,NBA=,OBA=,IMSID=,AGN=,
//      SSM=,PREINIT=,RGN=56K,SOUT=A,
//      SYS2=,ALTID=,APARM=,LOCKMAX=
//*
//G      EXEC PGM=DFSRR00,REGION=&RGN,
//      PARM=(BMP,&MBR,&PSB,&IN,&OUT,
//      &OPT&SPIE&TEST&DIRCA,&PRLD,
//      &STIMER,&CKPTID,&PARDLI,&CPUTIME,
//      &NBA,&OBA,&IMSID,&AGN,&SSM,
//      &PREINIT,&ALTID,
//      '&APARM',&LOCKMAX)
//STEPLIB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
//      DD DSN=IMS.&SYS2.PGMLIB,DISP=SHR
//PROCLIB DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR
//SYSUDUMP DD SYSOUT=&SOUT,
//      DCB=(LRECL=121,RECFM=VBA,BLKSIZE=3129),
//      SPACE=(125,(2500,100),RLSE,,ROUND)
```

Note: Parameters in parentheses are positional.

IMSBATCH Procedure Parameters

The following parameters are valid for the IMSBATCH procedure. See "Parameter Descriptions" on page 244.

AGN=	OUT=
ALTID=	PARDLI=
APARM=	PREINIT=
CKPTID=	PRLD=
CPUTIME=	PSB=
DIRCA=	RGN=
IMSID=	SOUT=
IN=	SPIE=
LOCKMAX=	SSM=

MBR=	STIMER=
MSG=	TEST=
NBA=	SYS2=
OBA=	
OPT=	

IMSBATCH Procedure DD Statements

The following DD statements are valid for the IMSBATCH procedure. See “DD Statement Descriptions” on page 279.

DFSCTL DD ¹	IMSLOGR DD ²
DFSESL DD ¹	PROCLIB DD
DFSSTAT DD ¹	STEPLIB DD
FPTRACE DD ¹	SYSUDUMP DD
IMS DD ¹	

Notes:

1. These DD statements are optional.
2. If the checkpoint records required to restart the BMP do not exist in the online log data sets (OLDSn), an //IMSLOGR DD statement specifying a data set containing the checkpoint log records must be added to the BMP job's JCL.

IMSCOBGO Procedure

Procedure Description

The following is a three-step compile, link-edit, and go procedure combining the procedure IMSCOBOL with an execution step for a stand-alone DL/I batch processing region.

If VSAM databases are used, see “Specifying IMS Buffer Pools” on page 378.

The following sample procedure assumes that:

- You supply source data from SYSIN.
- Output class is A.
- MBR=NAME, where NAME is the load module name for the program.
- SYSDA is a generic device name.

```
//      PROC MBR=TEMPNAME,PAGES=60,SYS2=,
//      SOUT=A,RGN=962K,LOGT=2400,
//      PSB=,BUF=7,SPIE=0,TEST=0,EXCPVR=0,
//      RST=0,PRLD=,SRCH=0,CKPTID=,MON=N,
//      LOGA=0,FMT0=T,IMSID=,SWAP=,DBRC=,IRLM=,
//      IRLMNM=,BKO=N,IOB=
//C      EXEC PGM=IKFCBL00,REGION=150K,
//      PARM='SIZE=130K,BUF=10K,LINECNT=50'
//SYSLIN DD DSN=&&LIN,DISP=(MOD,PASS),UNIT=SYSDA,
//      DCB=(IMS.&SYS2.PROCLIB),
//      SPACE=(3520,(40,10),RLSE,,ROUND)
//SYSPRINT DD SYSOUT=&SOUT,
//      DCB=(LRECL=121,BLKSIZE=605,RECFM=FBA),
//      SPACE=(605,(&PAGES.0,&PAGES),RLSE,,ROUND)
//SYSUT1 DD UNIT=SYSDA,DISP=(,DELETE),
//      SPACE=(3520,(100,10),RLSE,,ROUND)
//SYSUT2 DD UNIT=SYSDA,DISP=(,DELETE),
//      SPACE=(3520,(100,10),RLSE,,ROUND)
//SYSUT3 DD UNIT=SYSDA,DISP=(,DELETE),
```

IMSCOBGO Procedure

```
//          SPACE=(3520,(100,10),RLSE,,ROUND)
//SYSUT4   DD UNIT=SYSDA,DISP=(,DELETE),
//          SPACE=(3520,(100,10),RLSE,,ROUND)
//L        EXEC PGM=IEWL,REGION=150K,
//          PARM='XREF,LET,LIST',
//          COND=(4,LT,C)
//SYSLIB   DD DSN=SYS1.COBLIB,DISP=SHR
//RESLIB   DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
//SYSLIN   DD DSN=&&LIN,DISP=(OLD,DELETE),
//          VOL=REF=*.C.SYSLIN
//          DD DISP=SHR,
//          DSN=IMS.&SYS2.PROCLIB(CBLTDLI)
//          DD DDNAME=SYSIN
//SYSLMOD  DD DISP=SHR,
//          DSN=IMS.&SYS2.PGMLIB(&MBR)
//SYSPRINT DD SYSOUT=&SOUT,
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=605),
//          SPACE=(605,(&PAGES.0,&PAGES),RLSE,,ROUND)
//SYSUT1   DD UNIT=(SYSDA,SEP=(SYSLMOD,SYSLIN)),
//          DISP=(,DELETE),
//          SPACE=(3520,(100,10),RLSE,,ROUND)
//G        EXEC PGM=DFSRR00,REGION=&RGN,TIME=2,
//          COND=(4,LT),
//          PARM=(DLI,&MBR,&PSB,&BUF,
//          &SPIE&TEST&EXCPVR&RST,
//          &PRLD,&SRCH,&CKPTID,&MON,&LOGA,
//          &FMTO,&IMSID,&SWAP,
//          &DBRC,&IRLM,&IRLMNM,&BKO,&IOB)1
//STEPLIB  DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
//          DD DSN=IMS.&SYS2.PGMLIB,DISP=SHR
//DFSRESLB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
//IMS      DD DSN=IMS.&SYS2.PSBLIB,DISP=SHR
//          DD DSN=IMS.&SYS2.DBDLIB,DISP=SHR
//PROCLIB  DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR2
//IEFRDER  DD DSN=IMSLOG,DISP=(,KEEP),VOL=(,,99),
//          UNIT=(&LOGT,,DEFER),
//          DCB=(RECFM=VB,BLKSIZE=1408,LRECL=1400,BUFNO=2)3
//IEFRDER2 DD DSN=IMSLOG2,DISP=(,KEEP),VOL=(,,99),
//          UNIT=(&LOGT,,DEFER,SEP=IEFRDER),
//          DCB=(RECFM=VB,BLKSIZE=1408,
//          LRECL=1400,BUFNO=2)
//SYSOUT   DD SYSOUT=&SOUT,SPACE=(CYL,(1,1)),
//          DCB=(LRECL=133,RECFM=FBA,BLKSIZE=665)
//SYSUDUMP DD SYSOUT=&SOUT,
//          DCB=(LRECL=121,RECFM=FBA,BLKSIZE=3025),
//          SPACE=(3025,(200,100),RLSE,,ROUND)
```

Notes

1. Parameters in parentheses are positional.
2. IMS.PROCLIB should have the same block size as SYS1.PROCLIB, which should be 80, 400, or 3200.
3. The IEFRDER and IEFRDER2 statements are not required if the job does not declare database update intent.

For a job step declaring database-update intent, DD DUMMY can be specified if the job step is not using DBRC. This is valid where an image copy of the database is taken before the update job step.

Log initialization calculates the smallest value necessary for logical record length. If the JCL logical record length value is larger than the calculated value, the JCL value is used; otherwise, log initialization uses the calculated value for logical record length and adds 4 for the block size.

If multiple volumes are required for the system log, a volume count value should be specified in the VOL parameter of the DD statement.

If the IBM 3480 tape drive is used for the IMS log data set, IMS forces tape write mode (DCB=OPCD=W). The default on the 3480 is to hold the write in a buffer so that IMS cannot detect when the write is actually performed. If a power failure occurs after a log record is written to the 3480, and the database is updated but the log record is not yet written to tape, database integrity is lost. Tape write mode is forced for the log in batch and GSAM data sets.

IMSCOBGO Procedure Parameters

The following parameters are valid for the IMSCOBGO procedure. See “Parameter Descriptions” on page 244.

BKO=	MBR=
BUF=	MON=
CKPTID=	PRLD=
DBRC=	PSB=
EXCPVR=	RGN=
FMTO=	RST=
IMSID=	SOUT=
IOB=	SPIE=
IRLM=	SRCH=
IRLMNM=	SWAP=
LOGA=	SYS2=
LOGT=	TEST=

IMSCOBGO Procedure DD Statements

The following DD statements are valid for the IMSCOBGO procedure. see “DD Statement Descriptions” on page 279.

In addition to the following DD statements, add statements for data sets representing IMS databases.

DFSRESLB DD	STEPLIB DD
DFSSTAT DD	SYSLIB DD
IEFRDER DD	SYSLIN DD
IEFRDER2 DD	SYSLMOD DD
IMS DD	SYSOUT DD
PROCLIB DD	SYSPRINT DD
RECONn DD	SYSUDUMP DD
RESLIB DD	SYSUTn DD

IMSCOBOL Procedure

Procedure Description

The following is a two-step compile and link-edit procedure for IMS COBOL applications.

The following sample procedure assumes that:

- You supply source data from SYSIN.
- Output class is A.

IMSCOBOL Procedure

- MBR=NAME, where NAME is the load module name for the program.
- SYSDA is a generic device name.

```
//      PROC MBR=TEMPNAME,PAGES=60,SY2=,
//      SOUT=A
//C     EXEC PGM=IKFCBL00,REGION=150K,
//      PARM=' SIZE=130K,BUF=10K,LINECNT=50'
//SYSLIN DD DSN=&&LIN,DISP=(MOD,PASS),UNIT=SYSDA,
//      DCB=(IMS.&SY2.PROCLIB),
//      SPACE=(3520,(40,10),RLSE,,ROUND)
//SYSPRINT DD SYSOUT=&SOUT,
//      DCB=(LRECL=121,BLKSIZE=605,RECFM=FBA),
//      SPACE=(605,(&PAGES.0,&PAGES),RLSE,,ROUND)
//SYSUT1 DD UNIT=SYSDA,DISP=(,DELETE),
//      SPACE=(3520,(100,10),RLSE,,ROUND)
//SYSUT2 DD UNIT=SYSDA,DISP=(,DELETE),
//      SPACE=(3520,(100,10),RLSE,,ROUND)
//SYSUT3 DD UNIT=SYSDA,DISP=(,DELETE),
//      SPACE=(3520,(100,10),RLSE,,ROUND)
//SYSUT4 DD UNIT=SYSDA,DISP=(,DELETE),
//      SPACE=(3520,(100,10),RLSE,,ROUND)
//L     EXEC PGM=IEWL,REGION=150K,
//      PARM=' XREF,LET,LIST',
//      COND=(4,LT,C)
//SYSLIB DD DSN=SYS1.COBLIB,DISP=SHR
//RESLIB DD DSN=IMS.&SY2.RESLIB,DISP=SHR
//SYSLIN DD DSN=&&LIN,DISP=(OLD,DELETE),
//      VOL=REF=*.C.SYSLIN
//      DD DISP=SHR,
//      DSN=IMS.&SY2.PROCLIB(CBLTDLI)
//      DD DDNAME=SYSIN
//SYSLMOD DD DISP=SHR,
//      DSN=IMS.&SY2.PGMLIB(&MBR)
//SYSPRINT DD SYSOUT=&SOUT,
//      DCB=(RECFM=FBA,LRECL=121,BLKSIZE=605),
//      SPACE=(605,(&PAGES.0,&PAGES),RLSE,,ROUND)
//SYSUT1 DD UNIT=(SYSDA,SEP=(SYSLMOD,SYSLIN)),
//      DISP=(,DELETE),
//      SPACE=(3520,(100,10),RLSE,,ROUND)
```

IMSCOBOL Procedure Parameters

The following parameters are valid for the IMSCOBOL procedure. See “Parameter Descriptions” on page 244.

MBR= **SY2=**
SOUT=

IMSCOBOL Procedure DD Statements

The following DD statements are valid for the IMSCOBOL procedure. See “DD Statement Descriptions” on page 279.

RESLIB DD **SYSLMOD DD**
SYSLIB DD **SYSPRINT DD**
SYSLIN DD **SYSUTn DD**

IMSFP Procedure

Procedure Description

The following procedure executes a Fast Path application program.

Nonmessage-driven Fast Path applications are not supported and should be changed to run as BMPs, using the IMSBATCH procedure. See “IMSBATCH Procedure” on page 232

```
//IFPROC PROC MBR=TEMPNAME,SOUT=A,RGN=100K,OPT=N,
//          PSB=,NBA=000,OBA=000,TLIM=1,
//          DIRCA=000,PRLD=,STIMER=0,SOD=,DBLDL=,
//          CPUTIME=,IMSID=,AGN=,SSM=,PREINIT=,
//          SYS2=,ALTID=,APARM=,LOCKMAX=
//IFP EXEC PGM=DFSRR00,REGION=&RGN,
//        PARM=(IFP,&MBR,&PSB,&NBA,&OBA,&OPT,1
//        &TLIM,&DIRCA,&PRLD,&STIMER,&SOD,&DBLDL,
//        &CPUTIME,&IMSID,&AGN,&SSM,&PREINIT,
//        &ALTID,'&APARM',&LOCKMAX)
//STEPLIB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
//        DD DSN=IMS.&SYS2.PGMLIB,DISP=SHR
//PROCLIB DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR
//SYSUDUMP DD SYSOUT=&SOUT,
//        DCB=(LRECL=121,RECFM=VBA,BLKSIZE=3129),
//        SPACE=(125,(2500,100),RLSE,,ROUND)
```

Note:¹ These parameters are positional.

IMSFP Procedure Parameters

The following parameters are valid for the IMSFP procedure. See “Parameter Descriptions” on page 244.

AGN=	OPT=
ALOT=	PREINIT=
ALTID=	PRLD=
APARM=	PSB=
CPUTIME=	RGN=
DBLDL=	SOD=
DIRCA=	SOUT=
IMSID=	SSM=
LOCKMAX=	STIMER=
MBR=	SYS2=
NBA=	TLIM=
OBA=	

IMSFP Procedure DD Statements

The following DD statements are valid for the IMSFP procedure. See “DD Statement Descriptions” on page 279.

PROCLIB DD	SYSUDUMP DD
STEPLIB DD	

IMSMSG Job

Job Description

The following job executes procedure DFSMPR, which is used to start an online message region. See “DFSMPR Procedure” on page 211 for details.

IMSMSG Job

```
//MESSAGE JOB 1,IMS,MSGLEVEL=1,PRTY=11,CLASS=K,
MSGCLASS=A,REGION=56K
//      EXEC DFSMPR,
//      IMSID=
```

IMSPLI Procedure

The following procedure is a two-step compile and link-edit procedure for IMS PL/I application programs.

The following sample procedure assumes that:

- Your supply source data is from SYSIN.
- Output class is A.
- MBR=NAME, where NAME is the load module name for the program.
- SYSDA is a generic device name.

```
//      PROC MBR=TEMPNAME,PAGES=50,SYSDA=,
//      SOUT=A
//C      EXEC PGM=IEL0AA,REGION=114K,
//      PARM=(XREF,A,OBJ,NODECK,NOMACRO,,
//      'OPT(TIME)')
//SYSUT1 DD UNIT=SYSDA,
//      SPACE=(1024,(60,60),RLSE,,ROUND),
//      DCB=BLKSIZE=1024,DISP=(,DELETE)
//SYSPRINT DD SYSOUT=&SOUT,
//      DCB=(LRECL=125,BLKSIZE=629,RECFM=VBA),
//      SPACE=(605,(&PAGES.0,&PAGES),RLSE)
//SYSLIN DD UNIT=SYSDA,SPACE=(80,(250,80),RLSE),
//      DCB=(IMS.&SYS2.PROCLIB),
//      DISP=(,PASS)
//L      EXEC PGM=IEWL,PARM='XREF,LIST,LET',
//      COND=(4,LT,C),REGION=120K
//SYSLIB DD DSN=SYS1.PLIBASE,DISP=SHR
//RESLIB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
//SYSLIN DD DSN=*.C.SYSLIN,DISP=(OLD,DELETE)
//      DD DISP=SHR,
//      DSN=IMS.&SYS2.PROCLIB(PLITDLI)
//      DD DDNAME=SYSIN
//SYSLMOD DD DISP=SHR,
//      DSN=IMS.&SYS2.PGMLIB(&MBR)
//SYSPRINT DD SYSOUT=&SOUT,
//      DCB=(LRECL=121,RECFM=FBA,BLKSIZE=605),
//      SPACE=(605,(&PAGES.0,&PAGES),RLSE)
//SYSUT1 DD UNIT=SYSDA,DISP=(,DELETE),
//      SPACE=(CYL,(5,1),RLSE)
```

IMSPLI Procedure Parameters

The following parameters are valid for the IMSPLI procedure. See “Parameter Descriptions” on page 244.

MBR= **SYS2=**
SPIE=

IMSPLI Procedure DD Statements

The following DD statements are valid for the IMSPLI procedure. See “DD Statement Descriptions” on page 279.

RESLIB DD **SYSLMOD DD**
SYSLIB DD **SYSPRINT DD**

SYSLIN DD

SYSUT1 DD

IMSPLIGO Procedure

Procedure Description

The following procedure is a three step compile, link-edit, and go procedure combining the procedure IMSPLI with an execution step for a stand-alone DL/I batch processing region.

If VSAM databases are used, see “Specifying IMS Buffer Pools” on page 378.

The following sample procedure assumes that the following statements are true:

- You supply source data from SYSIN.
- Output class is A.
- MBR=NAME, where NAME is the load module name for the program.
- SYSDA is a generic device name.
- You add DD statements for data sets representing IMS databases.

The following is a sample of an IMSPLIGO procedure:

```
//      PROC MBR=TEMPNAME,PAGES=50,SYS2=,
//      SOUT=A,RGN=962K,
//      PSB=,BUF=7,SPIE=0,TEST=0,EXCPVR=0,
//      RST=0,PRLD=,SRCH=,CKPTID=,MON=N,
//      LOGA=0,FMT0=T,IMSID=,SWAP=,RGN=512K,
//      LOGT=2400,
//      DBRC=,IRLM=,IRLMNM=,BKO=N,IOB=
//C      EXEC PGM=IELOAA,REGION=114K,
//      PARM=(XREF,A,OBJ,NODECK,NOMACRO,,
//      'OPT(TIME)')
//SYSUT1 DD UNIT=SYSDA,
//      SPACE=(1024,(60,60),RLSE,,ROUND),
//      DCB=BLKSIZE=1024,DISP=(,DELETE)
//SYSPRINT DD SYSOUT=&SOUT,
//      DCB=(LRECL=125,BLKSIZE=629,RECFM=VBA),
//      SPACE=(605,(&PAGES.0,&PAGES),RLSE)
//SYSLIN DD UNIT=SYSDA,SPACE=(80,(250,80),RLSE),
//      DCB=(IMS.&SYS2.PROCLIB),
//      DISP=(,PASS)
//L      EXEC PGM=IEWL,PARM='XREF,LIST,LET',
//      COND=(4,LT,C),REGION=120K
//SYSLIB DD DSN=SYS1.PLIBASE,DISP=SHR
//RESLIB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
//SYSLIN DD DSN=*.C.SYSLIN,DISP=(OLD,DELETE)
//      DD DISP=SHR,
//      DSN=IMS.&SYS2.PROCLIB(PLITDLI)
//      DD DDNAME=SYSIN
//SYSLMOD DD DISP=SHR,
//      DSN=IMS.&SYS2.PGMLIB(&MBR)
//SYSPRINT DD SYSOUT=&SOUT,
//      DCB=(LRECL=121,RECFM=FBA,BLKSIZE=605),
//      SPACE=(605,(&PAGES.0,&PAGES),RLSE)
//SYSUT1 DD UNIT=SYSDA,DISP=(,DELETE),
//      SPACE=(CYL,(5,1),RLSE)
//G      EXEC PGM=DFSRR00,REGION=&RGN,TIME=5,
//      COND=(4,LT),
//      PARM=(DLI,&MBR,&PSB,&BUF,
//      &SPIE&TEST&EXCPVR&RST,
//      &PRLD,&SRCH,&CKPTID,&MON,&LOGA,
//      &FMT0,&IMSID,&SWAP,&DBRC,&IRLM,
```

IMSPLIGO Procedure

```
//          &IRLMNM,&BKO,&IOB) 1
//STEPLIB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
//          DD DSN=IMS.&SYS2.PGMLIB,DISP=SHR
//DFSRESLB DD DSN=IMS.&SYS2.RESLIB,DISP=SHR
//IMS      DD DSN=IMS.&SYS2.PSBLIB,DISP=SHR
//          DD DSN=IMS.&SYS2.DBDLIB,DISP=SHR
//PROCLIB  DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR 2
//IEFRDER  DD DSN=IMSLOG,DISP=(NEW,KEEP),
//          VOL=(, ,99),UNIT=(&LOGT, ,DEFER),
//          DCB=(RECFM=VB,BLKSIZE=1408,
//          LRECL=1400,BUFNO=2) 3
//IEFRDER2 DD DSN=IMSLOG2,DISP=(NEW,KEEP),
//          UNIT=(&LOGT, ,DEFER,SEP=IEFRDER),
//          VOL=(, ,99),DCB=(RECFM=VB,BLKSIZE=1408,
//          LRECL=1400,BUFNO=2) 3
//SYSPRINT DD SYSOUT=&SOUT,
//          DCB=(LRECL=121,BLKSIZE=605,RECFM=FBA),
//          SPACE=(605,(500,500),RLSE, ,ROUND)
//SYSUDUMP DD SYSOUT=&SOUT,
//          DCB=(LRECL=121,BLKSIZE=605,RECFM=FBA),
//          SPACE=(605,(500,500),RLSE, ,ROUND)
```

Notes:

1. These parameters are positional.
2. IMS.PROCLIB should have the same block size as SYS1.PROCLIB, which should be 80, 400, or 3200.
3. The IEFRDER and IEFRDER2 statements are not required if the job does not declare database-update intent.

For a job step declaring database-update intent, DD DUMMY can be specified if the job step is not using DBRC. This is valid where an image copy of the database is taken before the update job step.

The log-initialization process calculates the smallest value necessary for logical record length. If the JCL logical record length value is larger than the calculated value, the JCL value is used; otherwise, the log-initialization process uses the calculated value for logical record length and adds 4 for the block size.

If multiple volumes are required for the system log, a volume count value should be specified in the VOL parameter of the DD statement.

If the IBM 3480 tape drive is used for the IMS log data set, IMS forces tape write mode (DCB=OPCD=W). The default on the 3480 is to hold the write in a buffer so that IMS cannot detect when the write is actually performed. If a power failure occurs after a log record is written to the 3480, and the database is updated but the log record is not yet written to tape, database integrity is lost. Tape write mode is forced for the log in batch and GSAM data sets.

IMSPLIGO Procedure Parameter Descriptions

The following parameters are valid for the IMSPLIGO procedure. See "Parameter Descriptions" on page 244.

BKO=	MBR=
BUF=	MON=
CKPTID=	PRLD=
DBRC=	PSB=
EXCPVR=	RGN=
FMTO=	RST=
IMSID=	SOUT=

IOB=	SPIE=
IRLM=	SRCH=
IRLMNM=	SWAP=
LOGA=	SYS2=
LOGT=	TEST=

IMSPLIGO Procedure DD Statements

The following DD statements are valid for the IMSPLIGO procedure. See “DD Statement Descriptions” on page 279.

DFSRESLB DD	STEPLIB DD
DFSSTAT DD	SYSLIB DD
IEFRDER DD	SYSLIN DD
IEFRDER2 DD	SYSMOD DD
IMS DD	SYSPRINT DD
PROCLIB DD	SYSUDUMP DD
RECONn DD	SYSUT1 DD
RESLIB DD	

IMSRDR Procedure

Procedure Description

The following DASD procedure reads an IMSMSG job into the operating system job stream from direct access storage devices.

```
//          PROC MBR=IMSMSG,CLASS=A,SYS2=
//IEFPROC EXEC PGM=IEBEDIT
//SYSPRINT DD DUMMY
//SYSUT1   DD DDNAME=IEFRDER
//SYSUT2   DD SYSOUT=(&CLASS,INTRDR),DCB=BLKSIZE=80
//SYSIN    DD DUMMY
//IEFRDER  DD DISP=SHR,
//          DSN=IMS.&SYS2.JOBS(&MBR)
```

IMSRDR Procedure Parameter Descriptions

The following parameters are valid for the IMSRDR procedure. See “Parameter Descriptions” on page 244.

CLASS=	SYS2=
MBR=	

IMSRDR Procedure DD Statements

The following DD statements are valid for the IMSRDR procedure. See “DD Statement Descriptions” on page 279.

IEFRDER DD	SYSUT1 DD
SYSIN DD	SYSUT2 DD
SYSPRINT DD	

RDIBATCH Procedure

Procedure Description

The RDI region is used in a block-level data sharing environment to provide a path from the IRLM to DBRC to guarantee database availability if a failure occurs. The RDI region accomplishes the following:

- IDENTIFY to the IRLM
- SIGNON to DBRC
- WAIT for an IRLM failure.

If an IRLM fails, its associated RDI region abends with a U3303. The surviving IRLM partner is notified of the failure and, in turn, notifies DBRC of the failure.

The RDI region can only be terminated by an IRLM failure or by an operator CANCEL command. You must ensure that the IRLM and DBRC are both being used to gain the benefits of data sharing protection that this region offers.

The following procedure is **not** placed in IMS.PROCLIB by IMS system definition. You must create it by modifying a copy of either the DBBBATCH or DLIBATCH procedures. The region type is RDI.

Related Reading: If you are using IMS with CICS in an XRF environment, this procedure is essential. For more information about using IMS with CICS in and XRF environment, see *CICS Extended Recovery Facility Guide*.

```
//      PROC MBR=ANYNAME,1 PSB=,BUF=,
//          SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,
//          SRCH=0,CKPTID=,MON=,LOGA=0,FMTO=T,
//          IMSID=,SWAP=,DBRC=Y,2 IRLM=Y,3
//          IRLMNM=IRLM,4 BKO=N,IOB=,SSM=,APARM=,
//          RGN=2048K,
//          SOUT=A,
//          SYS=,
//          SYS2=,LOCKMAX=
//G      EXEC PGM=DFSRR00,REGION=&RGN,TIME=1440,5
//          PARM=(RDI,&MBR,&PSB,&BUF,
//              &SPIE&TEST&EXCPVR&RST,&PRLD,
//              &SRCH,&CKPTID,&MON,&LOGA,&FMTO,
//              &IMSID,&SWAP,&DBRC,&IRLM,&IRLMNM,
//              &BKO,&IOB,&SSM,
//              '&APARM',&LOCKMAX)
//STEPLIB DD DISP=SHR,DSN=IMS.&SYS2.RESLIB
//DFSRESLB DD DISP=SHR,DSN=IMS.&SYS2.RESLIB
//PROCLIB DD DISP=SHR,DSN=IMS.&SYS.PROCLIB
//RECON1 DD DISP=SHR,DSN=IMS.&SYS.RECON1
//RECON2 DD DISP=SHR,DSN=IMS.&SYS.RECON2
//RECON3 DD DISP=SHR,DSN=IMS.&SYS.RECON3
//SYSABEND DD SYSOUT=&SOUT.
          PEND
```

Notes:

1. MBR=xxxxxxx must be specified but does not need to be valid. The specified member will not receive control but must be specified or parameter analysis fails.
2. DBRC=Y is required (specified or used by default).
3. IRLM=Y is required (specified or used by default).
4. IRLMNM=cccc is required (specified or used by default).
5. TIME=1440 is recommended.

RDIBATCH Procedure Parameter Descriptions

The following parameters are valid for the RDIBATCH procedure. See “Parameter Descriptions” on page 244.

APARM=	MON=
BKO=	PRLD=
BUF=	PSB=
CKPTID=	RGN=
DBRC=	RST=
EXCPVR=	SOUT=
FMTO=	SPIE=
IMSID=	SRCH=
IOB=	SSM=
IRLM=	SWAP=
IRLMNM=	SYS=
LOCKMAX=	SYS2=
LOGA=	TEST=
MBR=	

RDIBATCH Procedure DD Statements

The following DD statements are valid for the RDIBATCH procedure. See “DD Statement Descriptions” on page 279.

DFSRESLB DD	STEPLIB DD
PROCLIB DD	SYSABEND DD
RECONn DD	

DL/I Interfaces

To establish a DL/I interface from COBOL or PL/I, use either the CBLTDLI procedure or the PLITDLI procedure.

CBLTDLI

The following control statements are necessary to establish a COBOL to DL/I interface. The block size of the following member must be less than or equal to 3200.

```
LIBRARY RESLIB(CBLTDLI)  DL/I LANGUAGE INTERFACE
LIBRARY RESLIB(DFHEI01) HLPI LANGUAGE INTERFACE
LIBRARY RESLIB(DFHEI1)  HLPI LANGUAGE INTERFACE
ENTRY DLITCBL
```

PLITDLI

The following control statements are necessary to establish a PL/I to DL/I interface. The blocksize of the following member must be less than or equal to 3200.

```
LIBRARY RESLIB(PLITDLI)  DL/I LANGUAGE INTERFACE
LIBRARY RESLIB(DFHEI01) HLPI LANGUAGE INTERFACE
LIBRARY RESLIB(DFHEI1)  HLPI LANGUAGE INTERFACE
ENTRY PLICALLA
```

PLITDLI is valid when using the PL/I Optimizing Compiler.

Parameter Descriptions

The following list contains a description of all the parameters that can be specified by the procedures described in this chapter.

AGN=

Specifies the one- to eight-character application group name used for interregion communication security. The default is blanks.

Related Reading: For details on the use of the application group name, refer to the Security Maintenance utility information in *IMS/ESA Administration Guide: System*.

ALOT=

Specifies the autologoff time in minutes. Valid values are 0 and from 10 to 1440. If the ALOT value is not specified, the value from the JCL member is used except for FINANCE, SLU P, and ISC. If ALOT is not specified on the logon descriptor or overridden by the logon exit (DFSLGNX0) for FINANCE, SLU P, and ISC, a value of 1440 is used (the value from the JCL member is ignored).

- **ALOT=0**

The terminal is logged off immediately when no signon is in effect. ALOT=0 is normally specified when signon occurs by one of the following methods:

- The user is signed on automatically during the logon process (autologon)
- Signon data is supplied by logon (BIND) userdata
- Signon data is supplied by the logon exit (DFSLGNX0)

Restriction: Do not specify ALOT=0 for interactive terminal sessions if the session expects a response to the DFS3649 (signon required) salutation message.

- **ALOT=(10-1439)**

The session is terminated after the specified number of minutes have elapsed without a signed-on user.

- **ALOT=1440**

The session is never automatically terminated.

ALOT only applies to ETO terminals.

ALTID=

Specifies a one- to four-character name for an alternate IMS system. When your IMS message processing region tries to link to an IMS system, it tries to link to the system specified by the IMSID= parameter. If no IMS system matches the name specified with IMSID, the message processing region tries the system specified with the ALTID= parameter. If no IMS system matches the name specified with ALTID=, what happens next depends upon the option you choose on the OPT= parameter.

AOIP=

Specifies a value for the upper expansion limit of the AOI buffer pool. The value can be specified as one- to six-numeric characters, or one- to five-numeric characters followed by K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value is 2G-1. If the specified value exceeds 2G-1, the default is 2G-1. If a value is not specified, the upper limit default is 2G-1.

AOIS=

Specifies whether RACF and the Command Authorization exit routine (DFSCCMD0) are to be used with AO applications for command security.

Related Reading: For more information, see "Establishing IMS Security" in the *IMS/ESA Administration Guide: System*. The possible values are:

- A** Includes options C and R below. RACF is called first. Then the security authorization facility (SAF) return code, RACF return code, and RACF reason code are passed to DFSCCMD0. These return codes are decoded into a security code, which is also passed to DFSCCMD0 for processing.
- C** Specifies that DFSCCMD0 is to be called for command authorization.
- N** Specifies that the DL/I call ICMD cannot be issued by an application program.
- R** Specifies that RACF is to be called for command authorization.
- S** Specifies that no authorization checking is to be done.

The default is N.

Because AOIS is not included in the checkpoint record, you can change the AOIS value each time IMS is initialized.

APARM=

Specifies the 1- to 32-character parameter that is passed to the application program as part of the information returned in the INQY call with the ENVIRON subfunction. The parameter must be enclosed in single-quotes (') if special characters are used. Embedded commas (,) are not allowed.

APPC=

Specifies to activate (Y) or not to activate (N) APPC/IMS LU 6.2 support. The default is (N). The /STA APPC command overrides APPC=N. An IMS APPLID override can also be specified via the existing APPLID1= execution parm during initialization.

APPCSE=

Specifies the type of APPC RACF security. The value you specify for this keyword has an equivalent value you can specify in the /SECURE APPC command. The first character you specify applies to the APPCSE keyword. The second character you specify becomes the equivalent value for the /SECURE APPC command.

The /SECURE APPC command overrides the value you specify in the APPCSE= keyword.

- | | | |
|----------|---------|---|
| C | CHECK | Specifies APPC RACF command and transaction security check. |
| F | FULL | Specifies APPC RACF command, transaction security check, and additional security for dependent regions. |
| N | NONE | Specifies no APPC RACF security within IMS. (Specifying NONE does not affect APPC/MVS RACF security.) |
| P | PROFILE | The IMS security exits (DFSCSTRN0 and DFSCCMD0) are still called if they are defined to the system.
Specifies the use of APPC transaction security in the TP profile or the default profile. |

The default is F.

Parameter Descriptions

APPLFE=

This optional parameter specifies a 1 to 8-character name of an application front-end routine that is called whenever a message processing program (MPP) is scheduled in a message processing region (MPR).

If APPLFE= is specified, the IMS MPP controller performs the following functions:

- Loads the application front-end routine.
- Calls the application front-end routine for initialization processing.
- Calls the application front-end routine when an MPP is scheduled in an MPR.
- Calls the application front-end routine for shutdown processing when the MPP terminates.

The following describes the interface between the IMS MPP controller and the application front-end routine:

1. The IMS MPP controller calls the application front-end routine for initialization processing with the input registers set as follows:

- R0=0
- R1=0
- R13=address of caller's save area
- R14=return address of caller
- R15=entry point address of the application front-end routine

The application front-end routine must restore all registers, except register 15, upon return to IMS. Register 15 contains the return code.

2. The IMS MPP controller calls the application front-end routine when an MPP is scheduled in an MPR with the input registers set as follows:

- R0=entry point address of the application MPP
- R1=PCB list to be passed to the application MPP
- R13=address of caller's save area
- R14=return address of caller
- R15=entry point address of the application front-end routine

The application front-end routine must restore all registers, except register 15, upon return to IMS. Register 15 contains the return code.

3. The IMS MPP controller calls the application front-end routine to perform shutdown processing when the MPP is terminating:

- R0=non-zero
- R1=0
- R13=address of caller's save area
- R14=return address of caller
- R15=entry point address of the application front-end routine

The application front-end routine must restore all registers, except register 15, upon return to IMS. Register 15 contains the return code.

Note: Because the IMS MPP controller establishes an ESTAE, a program interrupt is intercepted and treated as an application failure when it occurs after the application front-end routine is called and before it returns to the IMS MPP controller. As a result, the IMS MPP controller

Parameter Descriptions

reinitializes itself, and once again loads and calls the application front-end routine for initialization processing. IMS stops the transaction that is responsible for the interrupt.

APPLID1=name1

Specifies the application identification for the active IMS subsystem.

The VTAM APPLID is overridden by the APPLIDn= startup parameter, and APPLID3 is always used for the tracking subsystem. You must take care not to cause the same IMS VTAM APPLID to be opened for more than one IMS subsystem. If necessary, the APPLID=(*name1,name2,name3*) on the COMM macro should all be overridden by using the APPLID1, APPLID2, and APPLID3 parameters to ensure unique APPLIDS.

In a generic resource group environment, the APPLID1 parameter overrides the value specified in the first position of the APPLID= field of the IMS COMM macro. Using this execution parameter allows IMS sysgens to be identical by providing a unique APPLID override when starting the IMS job.

APPLID2=name2

Specifies the application identification for the XRF alternate IMS subsystem.

APPLID3=name3

Specifies the application identification for the RSR tracking IMS subsystem.

name3 can match either *name1* or *name2*, but no two application identifications can be active in the network at the same time.

ARC=

A non-zero entry (01-99) specifies that automatic archiving is to be performed. ARC= also indicates the number of online data sets (OLDSs) to be filled before issuing the DBRC GENJCL ARCHIVE command.

A /DBR command can cause automatic archiving, even if the number specified in ARC= is not reached.

IF ARC=00 is specified, no automatic archiving is performed.

The default is 01.

ARMRST=

Specifies whether (Y) or not (N) IMS will register with MVS ARM and allow ARM to restart IMS in case of a failure. The default is Y.

ASOT=

Specifies the autosignoff time, in minutes. Valid values are 0 and from 10 to 1440. You can override this value during signon with the user descriptor. If ASOT is not specified on either the user descriptor, the logon descriptor, or overridden by either the logon (DFSLGNX0) or signon (DFSSGNX0) exits, the value from the JCL member is used except for FINANCE, SLU P, and ISC. If ASOT is not specified on either the user or logon descriptor for FINANCE, SLU P, and ISC, a value of 1440 is used (the value from the JCL member is ignored).

- ASOT=0

The user is signed off immediately when no output is available to be sent. ASOT=0 is normally specified when no IMS input or output is available, or after the last available output message completes.

Parameter Descriptions

Restriction: Do not specify ASOT=0 for interactive terminals, for example 3270s or SLU2.

- ASOT=(10-1439)

The user is signed off after the specified number of minutes have elapsed without terminal activity, irrespective of message status.

- ASOT=1440

The user is never automatically signed off.

ASOT only applies to ETO terminals.

AUTO=

Specifies whether (Y) or not (N) automatic restart is to be invoked. The default is N.

BKO=

Specifies whether (Y) or not (N) dynamic backout is to be performed; BKO= applies only to pseudoabends. Dynamic backout is to the last sync point. Y only applies if the log data set is assigned to DASD. The default is N.

BSIZ=

Specifies the database buffer size; the value can be 512, 1024, 2048, 4096, or a multiple of 4K (4096). You must specify a value large enough to contain the largest defined DEDB control interval size. The maximum value is 28672.

BUF=

Specifies a one- to three-digit number defining the number of 1K blocks to be used in calculating the size of the OSAM subpools. This parameter only applies to a DL/I batch environment; it is used only if the DFSVSAMP data set is not supplied or does not include any IOBF control statements.

For more information, see "OSAM Buffer Pool Compatibility Definition" on page 380.

CHTS=

Specifies a one- to five-digit number of conversation hash table slots. Valid values are from 0 to 32767. The default is 256. If you specify an invalid or null value, 256 is assumed. For more information see "Specifying the Number of Hash Table Slots" on page 409.

CIOP=

Specifies a value for the upper expansion limit of the communication I/O buffer pool. The value can be specified as one- to six-numeric characters, or one- to five-numeric characters followed by K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value is 2G-1. If the value specified exceeds 2G-1, the default is 2G-1. If a value is not specified, the upper limit default is 2G-1.

CKPTID=

Specifies the checkpoint at which the program is to be restarted.

Enclose the checkpoint ID in single quotes if it contains any non-ANSI characters, such as the slash.

Related Reading: See "XRST Call" in *IMS/ESA Application Programming: Database Manager* for more information on specifying values for CKPTID.

You can suppress message DFS681I or DFS0540I each time a checkpoint call is made:

Parameter Descriptions

- To suppress message DFS6811, code NOMSG681.
- To suppress message DFS0540I, code NOMSG540.
- To suppress both messages, code NOMSGS.

The values listed above (NOMSG681, NOMSG540, and NOMSGS) are not valid checkpoint restart IDs.

CL1=,CL2=,CL3=,CL4=

Is a required positional parameter; it specifies four three-digit decimal numbers indicating which classes of messages are to be processed by this message region. For example, if classes 1, 2, and 3 are to be processed by this region, the PARM field must be specified as PARM='MSG,001002003000'.

The sequence of specifying the classes determines relative class priority within the message region. In the above example, all class 1 messages are selected for scheduling before any class 2 messages are considered. Class numbers cannot be greater than the maximum number of classes specified during system definition.

CLASS=

Specifies the default message class for the job being submitted to the internal reader.

CMDMCS=

The table below shows the procedures that describe the valid keywords for each online system type.

Table 19. Procedures for Online Systems

Online System Type	Procedure Name
DB/DC	IMS
DCCTL	DCC
DBCTL	DBC

The following sub-parameters apply to the IMS and DCC procedures.

- N** Commands cannot be entered from an MCS console. N is the default.
- Y** Commands can be entered from an MCS or E-MSC console by entering the command recognition character (CRC) followed by the command text.
- R** Commands can be entered from an MCS console in the form CRC followed by the command text. The system calls RACF (or equivalent) to verify that the user ID of the console is authorized to issue the command.
- C** Commands can be entered from an MCS console in the form CRC followed by the command text. DFSCCMD0 is called to verify that the user ID of the console is authorized to issue the command.
- B** Include options R and C. RACF (or equivalent) is called first. After RACF is called, the SAF return code, RACF return code and RACF reason code are passed to DFSCCMD0. The return codes are decoded into a security code that is also passed to DFSCCMD0. DFSCCMD0 determines if the command is authorized for the user ID of the console.

This is an optional sub-parameter. If you do not select B, C, or R, commands can be entered from an MCS console. Neither RACF nor DFSCCMD0 is called to verify that the user ID of the console is authorized to issue the command.

Parameter Descriptions

The following sub-parameters apply to the DBC procedure.

- R** RACF (or equivalent) is called to verify that the user ID of the console is authorized to issue the command.
- C** DFSCCMD0 is called to verify that the user ID of the console is authorized to issue the command.
- B** Include options R and C. RACF (or equivalent) is called first. After RACF is called, the SAF return code, RACF return code and RACF reason code are passed to DFSCCMD0. These return codes are decoded into a security code that is also passed to DFSCCMD0. DFSCCMD0 determines whether the command is authorized for the user ID of the console.

Restriction: CRC for DB/DC does not work for restart commands. The ability to issue IMS commands using the CRC or IMSID in a DB/DC environment is available if the proper specification of the CMDMCS parameter is supplied. However, this does not work for restart commands /ere and /nre. You must enter these commands the same as before. After restart is complete, you can enter commands using the CRC or the IMSID.

CORE=

This parameter is no longer used. CORE= has been replaced by the PIINCR and PIMAX execution parameters.

CPUTIME=

Specifies the task timing option for either BMP or Fast Path.

- 0** No task timing is performed for this BMP or Fast Path region. A default value is placed in the 07 accounting log record to indicate that no timing was done. This is the default.
- N** A one- to four-digit number specifying the maximum task time in minutes to be allowed for the execution of this BMP or Fast Path region. The values can range from 1 to 1440 minutes. If a value of 1440 or greater is specified, a time of 24 hours is used. When N is nonzero, the STIMER=1 option is in effect.

For BMP task-timing, the measured time includes DL/I processing occurring in the dependent region. When the dependent region exceeds the specified time, a U0240 abend occurs after the current DL/I call completes. To control the execution time of a BMP, use CPUTIME=N instead of the TIME= of the operating system. If the operating system time limit is included with a job that specifies CPUTIME=N, ensure that the former is a sufficiently higher value to allow for initialization and one completed DL/I call beyond the CPUTIME value. When the CPUTIME=N option is used, the application program must not use the STIMER or TTIMER CANCEL.

For Fast Path programs, the timer is set at the beginning of program invocation. However, message-driven programs are not allowed to exceed the amount of time defined for the program to process a single transaction. In this case, CPRTIME=*n* is ignored.

CRC=

CRC specifies a one-character command recognition character. Commands entered at an MVS system console with the CRC as a prefix, are processed by the DBCTL control region that has the CRC defined.

CRC overrides the value specified on the CMDCHAR keyword of the SYSGEN IMSCTRL macro.

Parameter Descriptions

If CMDCHAR is not specified, and CRC is not specified, the default value is /.

If CMDCHAR=NONE is specified, and CRC= is not specified, commands to the DBCTL control region are prefixed by the IMSID.

The CRC must be different from any other CRC in DBCTL regions already executing. Choose an unreserved character that is appropriate for your particular system. A comma can not be specified as a command recognition character.

Do not use a character for the CRC that is the same as the beginning character of an MVS command. If the CRC is the same as the beginning character of an MVS command, that command does not work after you start IMS. For example, if you start IMS with CRC=D, MVS does not respond to any of the MVS display commands such as D A,L.

The CRC value you select might affect multisegment command processing. The last character of the command is checked during multisegment command processing. If that character matches the CRC, the command is assumed to be a multisegment command. Processing of the command waits for the rest of the command. For example, if you use B for the command recognition character, and issue the command 'BMODIFY PREPARE ACBLIB', the command does not run. The ending 'B' in 'ACBLIB' is interpreted to indicate a multisegment command. To enter this command correctly, enter:

```
'BMODIFY PREPARE ACBLIB . '
```

Related Reading: Refer to *IMS/ESA Customization Guide* for more information on CRC.

CSAPSB=

Because the DL/I address space option (LSO=S) is used in the DBCTL environment, two program specification block pools are used. CSAPSB specifies the size of pool in the MVS common area, and DLIPSB specifies the size in the DL/I address space. In an LSO=S system, the PSB= parameter is ignored.

The output of the ACBGEN utility indicates maximum CSA space, average CSA space, and the CSA space required by each PSB in the pool. This output is useful in determining what value to specify for the CSAPSB= parameter. Similarly, before specifying the DLIPSB= parameter, you should consider the secondary address space (SAS) size required for each PSB, maximum SAS space, and average SAS space. Neither parameter should be zero. Normally, DLIPSB should be larger than CSAPSB.

The values for the pool sizes can be specified as either one- to six-numeric characters, or one- to five-numeric characters followed by either K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value is 2G-1. The upper limit defaults to 2G-1 if any of the following statements is true:

- The value specified is not large enough to hold the largest primary and secondary storage allocations that are defined.
- The value specified exceeds 2G-1.
- No value is specified.

Parameter Descriptions

The sizes specified are rounded up to the nearest page boundary. The sizes of the two pools can also be specified during IMS system generation, using the SASPSB parameter in the BUFPOOLS macro.

The sum of the values for the CSAPSB and DLIPSB defines the PSB pool size. If PSB is also specified, the larger value (PSB or the sum of CSAPSB and DLIPSB) is used.

For CSAPSB and PSBW, DFSIINS0 obtains contiguous space in the MVS common area. If this storage is unavailable in the MVS common area, ABENDU0717 occurs.

DBBF=

Specifies a one- to five-digit maximum number of buffers in ECSA for Fast Path databases. The default for this parameter is specified on the BFALLOC parameter of the FPCTRL macro during system definition; or the default of 10 is used if no value is specified. The value specified by this parameter can be increased, but not decreased, at emergency restart. The maximum value for DBBF= is 65535.

DBFX=

Specifies a one- to five-digit number of database buffers in ECSA that are to be available when the Fast Path dependent region is started.

DBD=

Specifies the Fast Path Data Entry Database (DEDB) DBD name. DBD= can be from one to eight characters.

DBLDL=

An optional four-digit number that specifies the maximum number of BLDL entries to be kept for application modules that are not preloaded. The default maximum is 20. The maximum value is 9999.

Related Reading: For more information, see references to BLDL in *IMS/ESA Administration Guide: System*.

DBRC=

Specifies whether database recovery control is to be used during this execution of IMS. The possible values for DBRC= are:

C This value has no meaning except during a batch backout run of IMS. If specified for other than a batch backout run, it is treated as a null.

With DBRC=C, you can back out batch jobs that terminate normally during an execution that includes DBRC **but not IRLM**. During batch backout of jobs that terminated normally, IMS performs authorization for databases and then backs out database changes, just as is done for jobs that terminate abnormally.

If DBRC=C is specified when the previous execution of IMS included both DBRC and IRLM, batch backout fails if the previous execution completed normally.

N Specifies that DBRC is not used during this execution unless DBRC=FORCE is specified on the IMSCTRL macro statement during IMS system definition. If DBRC=FORCE is specified during IMS system definition and if this is not a batch backout execution of IMS, a message is issued, and a nonzero return code is returned.

If this is a batch backout execution of IMS, the DBRC=FORCE system definition specification can be overridden by specifying DBRC=N on the

Parameter Descriptions

EXEC procedure. Thus, if the previous execution of IMS used DBRC but not IRLM, batch backout will execute without DBRC.

- Null** Specifies whether DBRC is included, based on the specification of the IMSCTRL macro statement keyword DBRC= during IMS system definition.
- Y** Specifies that DBRC is used during this execution of IMS.

If the IRLM is not being used, DBRC provides additional database security. If a common RECON data set is shared among subsystems, DBRC allows proper authorization to registered databases so that the subsystems can share data at the database level.

Attention: If DBRC is used to maintain the integrity of databases, DBRC=N should be specified only when DBRC is unavailable or when you are sure that the integrity of databases will not be destroyed. When DBRC=N, before reactivating DBRC, you must register information on the log volumes created while DBRC was inactive.

DBRCNM=

Specifies the member in SYS1.PROCLIB containing the procedure for the DBRC address space. The IMS control region automatically starts the DBRC procedure, using this specification. The name can be up to eight characters, the first of which must be alphabetic. The default is DBRC.

DBRSE=

Specifies the eight-character RSENAME. It is used by a primary and an alternate DBCTL region in a "DBCTL standby" configuration, and it must be the same in both DBCTL regions.

Related Reading: For more information on RSENAME, refer to *IMS/ESA Administration Guide: System*.

DBWP=

Specifies the amount of subpool 231 storage to be allocated to the database work area pool. The value can be specified as either one- to six-numeric characters, or one- to five-numeric characters followed by either K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value is 2G-1. If the value specified exceeds 2G-1, the default is 2G-1. The pool size specified is rounded up to the nearest page boundary.

The DBWP size must be specified as the largest logical record length parameter size of any database you might run.

DC=

Specifies the three-digit suffix for the DFSDCxxx member in the IMS.PROCLIB. The default is 000.

DFSDCxxx parameters allow the override of the VTAM node name and the LTERM names for the master and secondary terminals that are defined in the system generation.

DEADLOK=

Specifies two numbers in the form *iiii,kkkk* where:

- iiii* Specifies the number of seconds between IRLM local deadlock detection intervals. The default is 5.

Parameter Descriptions

kkkk Specifies the number of local deadlock cycles performed before global deadlock detection is performed. The default is 1.

The valid value for *iiii* is a number between 1 and 5. The value used for *kkkk* is always 1.

In data sharing environments, the IRLMs participating in data sharing synchronize their DEADLOK parameters to the values of the last IRLM connecting to the group.

Recommendation: Specify the same values for the DEADLOK parameter on all of your IRLM startup procedures if you don't want the value for the last IRLM to connect to the group to become the value for all.

DIRCA=

Specifies the size of the dependent region interregion communication area (DIRCA); the size specified must be a three-digit number (for example, 001) representing the number of 1-K blocks of subpool 251 to be reserved to hold a copy of your PCBs.

If you do not specify a DIRCA size, or if you specify a size of 000, IMS uses the system default for this message region. The size of the system default, determined during system initialization, is the maximum size required for any PSB. If necessary, IMS updates this default when an online change is made or when a dynamic PSB is scheduled.

Normally, you should not specify this parameter. You can use it to reduce the DIRCA size for a region where only PSBs with small DIRCA requirements are scheduled by class scheduling.

The output from message DFS6891 occurring during ACBGEN facilitates DIRCA size calculation. Each PSB has a PCB=value that is the sum of the PCBs within it. The PCB value plus 64 is the required DIRCA size, in bytes.

The Fast Path utility program uses a PSB that contains an I/O PCB and a DB PCB for the DBD specified in the JCL parameters. A DIRCA size of 002 is sufficient to contain the control blocks built for the PSB.

DLINM=

Specifies the member in SYS1.PROCLIB containing the procedure for the optional DL/I address space. If LSO=S is specified, DLINM= automatically starts the DL/I address space procedure. The name can be up to eight characters, the first of which must be alphabetic. The default is DLISAS.

DLIPSB=

Because the DL/I address space option (LSO=S) is used in the DBCTL environment, two program specification block pools are used. CSAPSB specifies the size of pool in the MVS common area, and DLIPSB specifies the size in the DL/I address space. In an LSO=S system, the PSB= parameter is ignored.

The output of the ACBGEN utility indicates maximum CSA space, average CSA space, and the CSA space required by each PSB in the pool. This output is useful in determining what value to specify for the CSAPSB= parameter. Similarly, before specifying the DLIPSB= parameter, you should consider the secondary address space (SAS) size required for each PSB, maximum SAS

space, and average SAS space. Neither parameter should be zero. Normally, DLIPSB should be larger than CSAPSB.

The values for the pool sizes can be specified as either one- to six-numeric characters, or one- to five-numeric characters followed by either K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value is 2G-1. The upper limit defaults to 2G-1 if any of the following statements is true:

- The value specified is not large enough to hold the largest primary and secondary storage allocations that are defined.
- The value specified exceeds 2G-1.
- No value is specified.

The sizes specified are rounded up to the nearest page boundary. The sizes of the two pools can also be specified during IMS system generation, using the SASPSB parameter in the BUFPOOLS macro.

For the FDR procedure, the sum of the values for the CSAPSB and DLIPSB defines the PSB pool size. If PSB is also specified, the larger value (PSB or the sum of CSAPSB and DLIPSB) is used.

DLQT=

Specifies a one- to three-digit number for the size of the dead letter queue, in days. Valid values are from 0 to 365. The default is 60. If you specify a null or invalid value, 60 is assumed.

DMB=

Specifies the amount of subpool 231 storage to be allocated to the DMB pool. The value can be specified as either one- to six-numeric characters, or one- to five-numeric characters followed by either K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value is 2G-1. If the value specified exceeds 2G-1, the default is 2G-1. The pool size specified is rounded up to the nearest page boundary.

The output of the ACBGEN utility indicates the size of each DMB processed. This output should be examined before specifying the DMB= parameter.

DPRTY=

Specifies the MVS dispatching priority at which the IMS control region should operate.

Related Reading: For details on DPRTY, see *MVS/ESA Job Control Language Reference*.

The IMS control region must not be executed at priority 0 or scheduled into a region whose priority falls within a JES2 APG. The control region's priority must be higher than an MVS APG if IMS message processing or batch message processing regions reside in the APG. A general rule to follow is that IMS CTL dispatching priority must always be higher than the dispatching priority of any IMS dependent region.

DSCT=

Specifies a one-character suffix identifying the user-supplied descriptor table, DFSDSCTy, in IMS.PROCLIB. The default is 0.

EMHB=

Specifies a value for the upper expansion limit of the Fast Path expedited message handler buffer pool. The value can be specified as one- to six-numeric

Parameter Descriptions

characters, or one- to five-numeric characters followed by K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G, is not specified, K is the default. The maximum value is 2G-1. The upper limit defaults to 2G-1 if any of the following statements is true:

- The value specified is not large enough to hold the largest primary and secondary storage allocations that are defined.
- The value specified exceeds 2G-1.
- No value is specified.

EMHL=

Specifies a one- to five-digit number for the length of the Fast Path expedited message handler buffer. If no buffer length is available from either the TRANSACT or APPL macro or, in the case of a static terminal, a FPBUF specification, this EMHL buffer length is used. Valid values are from 12 to 30720. The default is 2K. If you specify a null or invalid value, 2K is assumed.

EPCB=

Specifies the amount of subpool 231 storage to be allocated to the EPCB pool. The value can be specified as one- to six-numeric characters or one- to five-numeric characters followed by K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value that can be specified is 2G-1. If the value specified exceeds 2G-1 the default is 2G-1. The pool size specified is rounded up to the nearest page boundary. The size specified on the EPCB= parameter overrides the size specified in the BUFPOOLS macro.

If Fast Path is not generated, the EPCB pool is not allocated. If Fast Path is generated, an EPCB pool is required when the PSB, used by the IMS dependent region, has PCBs for Fast Path databases. This is true regardless of whether the region ever makes a call to these Fast Path databases. If nothing is specified on the EPCB= parameter or on the BUFPOOLS macro and if Fast Path is generated, the default pool size is 8K.

ETO=

Specifies whether extended terminal option is enabled. The default is N.

N Specifies that terminals that are not defined to IMS have session initiation attempts rejected. Logon user data is not supported for static terminals; any user data provided at logon time is ignored.

M Specifies that terminals that are not defined to IMS have session establishment attempts rejected. Logon user data is supported for static terminals; any user data provided at logon time allows automatic signon to occur during session initiation.

Y If ETO is requested during system definition, terminals not defined with system definition can establish sessions with IMS if both these conditions are true:

- The terminal type is supported by IMS for dynamic allocation.
- A default logon descriptor (provided either by IMS or by you) has sufficient information to create the control blocks necessary to accept the session request.

EXCPVR=

Specifies whether (1) or not (0) the OSAM Database Buffer Pool is to be page fixed. A value of 0 or 1 must appear in the generated JCL statement for this parameter.

EXVR=

Specifies whether (1 or Y) or not (0 or N) the Queue Manager buffer pools are to be long-term page fixed.

FBP=

Specifies the amount of subpool 0 storage to be allocated to the message format block pool. The value can be specified as one- to six-numeric characters or one- to five-numeric characters followed by K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G, is not specified, K is the default. The maximum value that can be specified is 2G-1. If the value specified exceeds 2G-1 the default is 2G-1. The pool size specified is rounded up to the nearest page boundary.

FDRMBR=

Specifies the two-digit suffix for the FDR member DFSFDRxx in the IMS.PROCLIB.

Restriction: When you specify the FDR parameter on the IMS and DBCTL procedures you must also specify IRLM=Y.

FESTIM=

Specifies a timeout value in seconds to be used for front-end switching. The minimum value is 1, the maximum value is 300. If this parameter is not specified, the default is the value specified at system definition.

FIX=

Specifies the two-character suffix for DFSFIXxx and DFSDRFxx. This specifies the IMS.PROCLIB member to control both of the following:

- Page fixing portions of the control program
- Loading portions of the control program into DREF storage

For information on defining the content of these members, see “Defining Fixed Pages in the IMS Control Region” on page 309.

FMTO=

Specifies the type of dump output to be produced.

For a detailed explanation of this parameter, see “Chapter 5. IMS Dumping and Dump Formatting Options” on page 287.

FPWP=

Specifies a value representing the upper expansion limit of the Fast Path work pool. The value can be specified as one- to six-numeric characters or one- to five-numeric characters followed by K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value that can be specified is 2G-1. If the value specified exceeds 2G-1 the default is 2G-1. If a value is not specified, the upper limit default is 2G-1.

FRE=

Specifies a one- to five-digit number of fetch request elements used to load MFS blocks into the message format block pool.

IRLMGRP=

Specifies the name of the XCF group to which this IRLM belongs. All IRLMs in the same group must specify the same LOCKTABL parameter and an IRLMID value that is unique within the group. The default is IRLMDS.

The group name (IRLMDS by default) is used as the XCF group name. This name cannot start with “SYS” and cannot be the same as the LOCKTABL parameter.

Parameter Descriptions

GRNAME=

Specifies a one- to eight-character name of the XCF group for IMS Open Transaction Manager Access (OTMA). The group name should be alphanumeric (upper case only) or specified with the following special characters: #, \$, @.

The member name IMS uses when joining the XCF group comes from the USERVAR specification or defaults to the IMS APPLID specification.

This parameter is only valid in IMS DB/DC or IMS TM-DB2 environments.

GRSNAME

Specifies a one- to eight-byte character name of the generic resources group. The group name should be specified in upper case alphanumeric characters, or with the following special characters: #, \$, @. GRSNAME must begin with an alphabetic character. You must specify the same GRSNAME for all IMS systems participating within the same generic resources group.

GSGNAME=

Specifies the global service group name to be used. If GSGNAME= is not specified in the DBBATCH procedure, the default is the GSGNAME specified in the IMSCTRL macro.

If GSGNAME=NONE is specified, this procedure does not reference any global service group and any activity performed is not tracked by RSR.

If a GSGNAME is supplied either by the GSGNAME= specification or from the system definition IMSCTRL macro, then *DBRC=Y* must be used.

HIOP=

Specifies a value for the upper expansion limit of the high communication I/O pool. The value can be specified as one- to six-numeric characters or one- to five-numeric characters followed by K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value that can be specified is 2G-1. If the value specified exceeds 2G-1, the default is 2G-1. If a value is not specified, the upper limit default is 2G-1.

HSBID=

Specifies (for XRF systems only) the XRF system identification. One system is associated with digit 1 and one is associated with digit 2. Either can be brought up as an active or an alternate system.

Through this parameter, IMS identifies the master terminal and the message queues that are associated with it. For example, an IMS system with HSBID=1 uses the first master terminal in the stage 1 definition.

It is recommended that you code the HSBID parameter if you are planning to run XRF anytime in the future. Having HSBID coded could save performing an IMS generation at a later date.

XRF requires that two APPLIDs be specified on the system definition COMM macro. XRF also requires that two *nodenames* be specified by the NAME parameter on the TERMINAL macro. These are for the master and secondary master terminal.

If a null value is specified for HSBID (for example, HSBID= ' '), XRF is disabled. Changing the HSBID from usable ('1' or '2') to null or from null to usable requires a cold start.

HSBMBR=

Specifies (for XRF systems only) the two-digit suffix for XRF member DFSHSBxx in the IMS.PROCLIB. The default is 00.

IMSGROUP=

Specifies a one- to four-character IMS group name. This parameter is user-generated and is usually the IMSID value most often found in the BMP or other dependent region PROCLIB members. There is no default value. If no IMS group name is specified, the MVS name token is *not* built.

The IMSGROUP= parameter must equal the same value as the IMSID of the BMP regions.

The IMSID of the CTL region should be different because it must be unique within the sysplex. Using the IMSID of the BMP regions as the IMSGROUP= value in the CTL region is to eliminate the need to add any new parameters to the BMP (and other dependent region's) JCL.

Example:

```
CTL Region 1 IMSID=IMS1, IMSGROUP=IMSB
CTL Region 2 IMSID=IMS2, IMSGROUP=IMSB
BMP Region 1 IMSID=IMSB
MPP Region 1 IMSID=IMSB
```

In the above example, BMP region 1 connects to either IMS1 or 2 using MVS names services to find the correct IMSID to use. The IMSID of the BMP JCL is used to construct the MVS name. This example also holds true for the MPP region and any IFP regions.

The one EXEC parameter, IMSGROUP= is added to the control region procedure.

IMSID=

Specifies a one- to four-character identifier that is a valid subsystem identifier to the operating system being used. This identifier is used instead of the identifier specified at system definition of the IMS system being executed.

The IMSID name must not be the same as the procedure name that starts IMS unless one of the following is true:

- All DD statements in the startup procedure are cataloged in the master catalog.
- The unit and volume are specified on each DD statement.

This parameter cannot be changed at emergency restart.

An IMSID name specified in the FDR procedure identifies the subsystem name for the FDR region.

IN=

Specifies an input transaction code. This parameter is necessary only when the application program intends to access the message queues. If this parameter is specified, the OUT= parameter is ignored. You cannot schedule a BMP with an IN= parameter specified against a PDIR with an associated SMB that is already scheduled. Doing so results in ABENDU0452 when the BMP is scheduled.

IOB=

This parameter is no longer used. Requests for I/O are now dynamically allocated.

Parameter Descriptions

IRLM=

Specifies whether (Y) or not (N) you want to use the IRLM during this execution.

The default for IRLM= depends on the values you specify in the IRLM= and the IRLMNM= parameters of the IMSCTRL macro. The default is IRLM=N under either of the following conditions:

- The IMSCTRL macro specifies IRLM=N.
- The IRLM= and IRLMNM= keywords of the IMSCTRL macro do not specify a value.

The default is IRLM=Y under either of the following conditions:

- The IMSCTRL macro specifies IRLM=Y.
- The IMSCTRL macro specifies an IRLMNM and IRLM=N is **not** specified.

If IRLMNM= is not specified on the IMSCTRL macro or on the execution JCL and IRLM= is specified on the execution JCL, then the IRLM name used is *IRLM*.

This parameter cannot be changed at emergency restart.

IRLMID=

Specifies a decimal number that is used to distinguish between IRLMs in a data sharing group. The IRLM with the lowest ID number in the group becomes the global deadlock manager for the group when you are in a data sharing module. There is no default value. A unique value must be specified for each IRLM in the data sharing group.

You can specify this parameter as either a one- to three-digit number from 1 to 255, or as a printable character. You must enclose a printable character with seven single quotation marks on either side of the character. Thus, you must specify the character *D* as *IRLMID='''D'''*. When you specify a printable character, IRLM uses the EBCDIC value of the character as the IRLMID.

IRLMNM=

Specifies the four-byte MVS subsystem name assigned to this IRLM. IRLM requires a four-byte name for its internal processing, even though MVS can accept names with fewer than four bytes for subsystem names.

ISIS=

Specifies whether resource access security checking is to be performed.

- 0** Specifies that no resource access security is to be performed.
- 1** Specifies that resource access security checking using RACF is to be performed.
- 2** Specifies that resource access security checking using the user-written exit routine is to be performed.

This parameter cannot be changed at emergency restart.

LGMSGSZ=

Specifies a one- to five-digit number that represents the size in bytes of a long message record. The length specified is rounded up to the nearest multiple of four.

Restriction: The length must be greater than or equal to the length you specify for the short message record (specified with SHMSGSZ=). If you specify a

Parameter Descriptions

value less than the short message record length, the length is obtained from whichever is greater: the DCB generated for the long message queue data set or the short message record length. The length of the long message record cannot exceed 30632 bytes. If you specify a value greater than 30632, the size is set to 30632.

This parameter is only valid in the Shared Queues environment. If you do not specify a value for the SHAREDQ= parameter, the value specified for LGMSGSZ= is ignored. If you specify a value for the SHAREDQ= parameter, but not for the LGMSGSZ= parameter, the size of the long message record is obtained from whichever is greater: the DCB generated for the long message queue data set or the short message record length.

LGNR=

Specifies the maximum number of Fast Path DEDB buffer alterations that is to be held before the entire control interval is logged. The parameter is used by Fast Path to build the Fast Path buffer header (DMHR) and to control the DEDB update logging mechanism. The number of DEDB buffer alterations has no direct relationship to the number of DEDB calls. The parameter can be specified as a one- to two-digit numeric value. If a value less than seven is specified, a minimum value of seven is set.

LHTS=

Specifies a one- to five-digit number of LTERM hash table slots. Valid values are from 0 to 32767. The default is 256. If you specify a null or invalid value, 256 is assumed. For more information see "Specifying the Number of Hash Table Slots" on page 409.

LOCKMAX=

Specifies a value between 1- and 32767- (in units of 1000). This parameter overrides the PSBGEN LOCKMAX value if one was specified. An override parameter of LOCKMAX=0 turns off all locking limitation.

LOCKMAX=10

Allows for 10000 locks.

LOCKMAX=0

Turns off locking limitation.

LOCKTAB=

This parameter is optional and specifies the lock table to be used by this group. The lock table must have previously been defined through XCF Resource Manager panels and must be an element of the currently active locking policy. The default is IRLMLT1.

This parameter must be the same for all IRLMs that specify the same value for the GROUP parameter.

LOGA=

This parameter is no longer used and is ignored. In earlier releases, it specified whether IMS was to use the BSAM (0) or OSAM (1) logging access method.

LOGT=

Specifies the tape device type where the log data set is to be mounted. The tape device type specified by the LOGT= parameter substitutes for the device parameter specified in the IEFORDER DD statement. The default is device type 2400.

LTERM=

Specifies whether the LTERM name of the static input terminal is used in the

Parameter Descriptions

DFSAPPC. Y the default, specifies the LTERM is used. N specifies the LTERM is not used. If LTERM=N is specified, and a user ID is not provided or a user is not signed on, the DFSAPPC process is rejected with an error message DFS1957E.

LSO=

Specifies whether the Local Storage Option is to be used. When it is used, some IMS modules and buffers are moved from the CSA to the private storage area of the control region.

Y Specifies that LSO is to be used. This is the default.

S Specifies the DL/I subordinate address space option. This is required for a CCTL connected to an IMS control region or DBCTL region.

If LSO=Y is specified for an RSR tracking subsystem, DL/I database tracking is not initialized; only Fast Path database tracking can be performed.

LUMC=

Specifies a value for the upper expansion limit of the LU 6.2 device manager common buffer pool. The value can be specified as one- to six-numeric characters or one- to five-numeric characters followed by K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value is 2G-1. The upper limit defaults to 2G-1 if any of the following statements is true:

- The value specified is not large enough to hold the largest primary and secondary storage allocations that are defined.
- The value specified exceeds 2G-1.
- No value is specified.

The OCO LUM code uses the pool for work area processing.

LUMP=

Specifies value for the upper expansion limit of the LU 6.2 device manager private buffer pool. The value can be specified as one- to six-numeric characters, or one- to five-numeric characters followed by K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value is 2G-1. The upper limit defaults to 2G-1 if any of the following statements is true:

- The value specified is not large enough to hold the largest primary and secondary storage allocations that are defined.
- The value specified exceeds 2G-1.
- No value is specified.

The OCO LUM code uses the pool for work area processing when not processing in the dependent region.

MAXCSA=

The DXRJPROC sample member has a DEFAULT of 6M. The acceptable value range is 1M - 999M. Specify the maximum amount of common service area (CSA) and extended CSA (ECSA) that the IRLM for this IMS uses for its lock control block structures. IRLM is not prevented from using additional CSA and EC SA for other purposes. You can enter the value in bytes (such as 5242880), or use the abbreviations K for kilobytes (such as 5000K) and M for megabytes (such as 5M).

MAXPST=

Specifies the maximum number of blocks that the user permits the online control region to allocate. The default is 255; the maximum is 999. When

system activity increases, IMS allocates PST blocks up to the maximum specified. When the workload diminishes, these blocks are unallocated and storage is released to as low as the value specified by PST=. An example of how these parameters can be specified follows:

Example:

```
...,PARM='PST=48,MAXPST=300,...
```

MAXUSRS=

Specifies the maximum number of IRLMs that are to connect to the data sharing group. A value from 2 to 248 can be specified.

The initial allocation of locking resources in XES is dependent on the number of entries and the maximum number of users, so some assessment of available resources is made when IRLM joins the group. However, nothing is done to keep the number of users below the specified value.

MBR=

Specifies an application program name.

MON=

Specifies whether (Y) or not (N) the IMS monitor is to be active for this execution.

MSDB=

Specifies the one-character suffix for DBFMSDBx. (See "Member DBFMSDBx" on page 304.)

MSG=

Is a required positional parameter; it indicates that message region is to be started.

NBA=

A four-digit number specifying the number of Fast Path database buffers to be made available in the Common Service Area when the Fast Path region is activated. These buffers are used by the DEDBs to access VSAM control intervals, and by the MSDBs and DEDBs to hold information for updating between the DL/I call and synchronization point.

These buffers become part of the system database resources, but the number specified is reserved for exclusive use by this region when the application program accesses the MSDBs or DEDBs. (Although you are guaranteed access to a number of buffers, a maximum of the NBA + OBA value, you might have to wait for them.) The default is 0.

NHTS=

Specifies a one- to five-digit number of terminal hash table slots. Valid values are from 0 to 32767. The default is 256. If you specify a null or invalid value, 256 is assumed. For more information see "Specifying the Number of Hash Table Slots" on page 409.

NLXB=

Specifies the number of parallel sessions to be added during the startup of an IMS system. The NLXB value is added to the value specified in the SESSION parameter during system definition to increase the number of link extension blocks (LXB) generated for each link control block (LCB). The default for NLXB is 0. If NLXB is specified when the SESSION value is already set at the maximum (676), the NLXB value is ignored. This parameter is valid for MSC with VTAM only and does not apply to Intersystem Communication (ISC).

Parameter Descriptions

OBA=

A four-digit number specifying the number of additional page-fixed buffers to be made available to a Fast Path application region if the normal allotment is used. In the Common Service Area, the system page fixes only enough additional buffers to handle the largest overflow specification from all active regions. The default is 0.

The system only allows one program at a time to use its overflow allocation. This is done to prevent a situation in which every program is waiting for buffers, when none is available. If an application needs more buffers than the sum of NBA + OBA, it abnormally terminates (for message-driven programs) or an FR status code is issued (for nonmessage-driven programs).

OPT=

Specifies what action to take when the message processing region starts, but when no system identifier matches the names given on IMSID or ALTID. The actions are:

- W** Wait for the control program to start.
- N** Ask operator for a decision. This is the default.
- C** Cancel message region automatically.

OTHR=

Specifies the number of concurrent output threads that Fast Path is to support for the entire Fast Path system. This can be any value from 1 to 255, but no more than the value specified on the MAXPST= parameter of the IMS procedure. OTHR= *n* causes a total of *n* service request blocks (SRBs) and extended service request blocks (ESRBs) to be created at system initialization. These blocks are used in scheduling asynchronous DEDB output. If an insufficient number of SRBs is specified, write buffers are queued until one becomes available.

OTMA=

Specifies that the IMS Open Transaction Manager Access (OTMA) function is to be enabled during IMS initialization. Valid values are Y(yes) or N(no). The default value is N.

If Y is specified, IMS attempts to create the OTMA group during initialization and then attempts to join that group.

This parameter is only valid in IMS DB/DC or DCCTL environments.

OTMAMD=

Specifies whether the member override function of the OTMA Prerouting Exit routine (DFSYPX0) is enabled for a transaction initiated from an OTMA client. Valid values are Y(yes) or N(no). The default value is N.

OTMANM=

Specifies the member name IMS uses when joining the XCF group for non-RSR or non-XRF systems.

OTMASE=

Specifies the type of OTMA RACF security. The following values are possible:

- C** OTMA RACF security is CHECK. The existing RACF calls are made. IMS commands are checked against the CIMS class. IMS transactions are checked against the TIMS class.

Parameter Descriptions

- F** OTMA RACF security is FULL. This is the same as CHECK, but additional security checking is performed on dependent regions.
- N** OTMA RACF security is NONE. No calls to RACF are made.
- P** OTMA RACF security is PROFILE. Each OTMA message defines the level of security checking to be done.

The /SECURE OTMA command overrides the value you specify in the OTMASE= keyword.

The default value is F.

OTMASP=

Specifies whether a non-synchronous Tpipe or a synchronous Tpipe will be created to deliver the OTMA output. Valid values are Y(create a synchronous Tpipe) or N(create a non-synchronous Tpipe). The default value is N. If the value of this parameter is Y or if the OTMA Prerouting Exit routine (DFSYDRU0) indicates the need for a synchronous Tpipe, the synchronous Tpipe will be created.

OUT=

Specifies the transaction code or logical terminal name to which an output message is to be sent. This parameter is necessary when the application program wants to send output without accessing the input queues. A remote LTERM (CNT) cannot be specified.

This parameter is ignored if IN= is also specified.

OVLA=

Specifies the overlay supervisor option:

- 0** Allow MVS to load and delete the overlay supervisor for every overlay application program. This is the default.
- 1** Load and retain a copy of the overlay supervisor when the message region is initialized.

PAGES=

Specifies the amount of primary and secondary space allocated for the SYSPRINT DD statement data set.

PARDLI=

Specifies the parallel DL/I option.

- 0** DL/I processing is to be performed within the BMP region. This is the default.
- 1** All DL/I processing for this BMP is to be performed in the IMS control region.

PARDLI=1 prevents control region system 113 abends resulting from system X22 abends in the BMP region. If PARDLI=1, parallel DL/I is disabled. This can degrade performance.

PC=

If PC=YES is specified, the IRLM lock control block structures reside in the IRLM address space, and the space switching program call (PC) instruction is used to obtain addressability to the structures. The MAXCSA value is ignored.

If PC= NO is specified, the IRLM lock structures reside in ECSA. The default for this parameter is PC=NO. The MAXCSA value is used to limit the amount of

Parameter Descriptions

CSA + ECSA used by IRLM for lock structures, and is considered 'accountable' storage in the STATUS display. However, IRLM can use CSA & ECSA beyond this value for other required structures. This is considered 'non-accountable' in the STATUS display.

If the value NO is specified, the IRLM lock structure resides in CSA. The default for this parameter is PC=NO.

PCB=

Specifies the size of the dependent region interregion communication area (DIRCA); the size specified must be a three-digit number (for example, *001*) representing the number of 1 KB blocks of subpool 251 to be reserved to hold a copy of your PCBs.

If you do not specify a DIRCA size or if you specify a size of *000*, IMS uses the system default for this message region. The size of the system default, determined during system initialization, is the maximum size required for any PSB. If necessary, this system default is updated when an online change is made or when a dynamic PSB is scheduled.

If you specify this parameter, the specified size is always used. If the size required is larger than the specified size, the region abends with a user 0242 abend code.

Normally, you should not specify this parameter. You can use it to reduce the DIRCA size for a region when PSBs with small DIRCA requirements are scheduled into this region.

The output from message DFS589I occurring during ACBGEN facilitates DIRCA size calculation. Each PSB has a PCB= value that is the sum of the PCBs within it. The PCB value + 64 becomes the required DIRCA in bytes.

PIINCR=

Specifies the increment to be used in a conditional GETMAIN to obtain the dynamic storage. The default value is 64K.

No storage is allocated for ENQUEUE/DEQUEUE blocks at initialization. The first block of storage, obtained when the ENQUEUE/DEQUEUE routine is first called, is in the increment specified in PIINCR. Storage continues to be obtained in the increment specified until the maximum is reached (see PIMAX=). IMS does not release any storage that is dynamically obtained for the ENQUEUE/DEQUEUE routine. After storage is obtained, it is kept for the duration of the control region execution.

PIINCR can be specified as either one- to six-numeric characters, or one- to five-numeric characters followed by K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value is 2G-1. The upper limit defaults to 2G-1 if any of the following statements is true:

- The value specified is not large enough to hold the largest primary and secondary storage allocations that are defined.
- The value specified exceeds 2G-1.
- No value is specified.

PIMAX=

Specifies the maximum amount of dynamic storage available to the exclusive control of the ENQUEUE/DEQUEUE routine. The default value is 1024K (1M).

Parameter Descriptions

No storage is allocated for ENQUEUE/DEQUEUE blocks at initialization. The first block of storage, obtained when the ENQUEUE/DEQUEUE routine is first called, is in the increment specified by PIINCR=. Storage continues to be obtained in the increment specified until the maximum is reached. IMS does not release any storage that is dynamically obtained for the ENQUEUE/DEQUEUE routine. After storage is obtained, it is kept for the duration of the control region execution.

PIMAX can be specified as either one- to six-numeric characters, or one- to five-numeric characters followed by either K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value is 2G-1. The upper limit defaults to 2G-1 if any of the following statements is true:

- The value specified is not large enough to hold the largest primary and secondary storage allocations that are defined.
- The value specified exceeds 2G-1.
- No value is specified.

PRDR=

Specifies the name of the IMSRDR procedure in IMS.PROCLIB, or SYS1.PROCLIB, used in /START REGION commands. The name can be up to eight characters in length. The default is IMSRDR.

PREINIT=

Specifies a two-character suffix for DFSINTxx, the IMS.PROCLIB member that lists the preinitialization modules that are to receive control. For details on how to define the member, see “Chapter 7. Tailoring the IMS System to Your Environment” on page 299.

PREMSG=

Specifies whether, (Y) or (N), you want to receive or suppress the DFS000I prefix message that precedes all DBCTL system messages and command responses.

If the DFS000I prefix message is suppressed (PREMSG=N), all system messages are issued with the IMSID appended to the last line. The message can be a single line with a maximum of 121 characters of message text followed by XXXX, where XXXX is the IMSID. Or the message can be multiple lines with each line containing a maximum of 71 characters of message text. The IMSID will be appended to the last line of a multi-line message or will be the only text of a final line.

Display command output is prefaced by a prefix message:

```
DFS4444I DISPLAY FROM ID=XXXX
```

where XXXX is the IMSID. The message text is in columns 2-71 of all subsequent lines. Column 1 of display output contains + if the message text continues to the next line; otherwise, column 1 is blank.

If PREMSG=Y (the default value) is selected, all system messages and command responses are issued as multi-line messages.

The first line is:

```
DFS000I MESSAGE(S) FROM ID=XXXX
```

where XXXX is the IMSID. The message starts on the second line.

Parameter Descriptions

PRLD=

Specifies a two-character suffix for DFSMPLxx, the IMS.PROCLIB member identifying the modules to be preloaded in the region or partition. For more information, refer to "Member DFSMPLxx" on page 318 and "Making High-Use Program Modules Resident" on page 318.

PSB=

Restriction: The following description applies to batch procedures only.

Is an optional parameter specifying a PSB name when the PSB name and application program name are different.

The use of generated PSBs (GPSB) is supported in a TM batch environment. If the application program requires a GPSB, it must use the PSB parameter to indicate that request. In this case the PSB parameter does not specify the name of the PSB. Instead, it specifies a code that requests the use of a generated PSB and its language type. The specification of the code as the PSB parameter causes the specified GPSB to be utilized for the batch application program. The application program name (MBR parameter) is used as the GPSB name. GPSBs are not available in DB batch.

The following coded character strings are used to identify the use of a GPSB and its language:

DF\$\$\$\$ASM GPSB for the assembler language format.

DF\$\$\$\$COB GPSB for the COBOL language format.

DF\$\$\$\$PLI GPSB for the PL/I language format.

DF\$\$\$\$PAS GPSB for the Pascal language format.

PSB=

Restriction: The following definition applies to online procedures only.

Specifies the amount of subpool 231 storage to be allocated to the PSB pool. The value can be specified as one-to-six numeric characters or one-to-five numeric characters followed by K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value that can be specified is 2G-1. If the value specified exceeds 2G-1 the default is 2G-1. The value specified is rounded up to the nearest page boundary.

The output of the ACBGEN utility indicates the maximum PSB size and the size of each PSB processed. This output should be examined before specifying the PSB= parameter.

For the FDR procedure, the sum of the values for the CSAPSB and DLIPSB defines the PSB pool size. If PSB is also specified, the larger value (PSB or the sum of CSAPSB and DLIPSB) is used.

PSBW=

Specifies the amount of subpool 231 storage to be allocated to the PSB work area pool. The value can be specified as one to six-numeric characters or one to five-numeric characters followed by either K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value that can be specified is 2G-1. If the value specified exceeds 2G-1 the default is 2G-1. The pool size specified is rounded up to the nearest page boundary.

Parameter Descriptions

The output of the ACBGEN utility indicates the maximum work area size and the work area size required by each PSB. This output should be examined before specifying the PSBW= parameter.

This parameter is not needed for an RSR tracking subsystem.

For CSAPSB and PSBW, DFSIINS0 obtains contiguous space in the MVS common area. If this storage is unavailable in the MVS common area, ABENDU0717 occurs.

PST=

Specifies a one- to three-digit number of PSTs (partition specification tables) to be allocated during system initialization. This number specifies the minimum number of dependent regions that are to be active. If more dependent regions are required during peak periods, IMS dynamically allocates more PSTs as additional dependent regions are started. PSTs can be allocated up to the MAXPST value. During idle periods, the PSTs are released. To the number specified on PST= is maintained. For example, if PST=15, it is assumed approximately 15 dependent regions are always active. During the peak period, if more dependent regions are started, IMS dynamically allocates a PST for each dependent region, up the MAXPST value. As PSTs are terminated, the PSTs are released until a maximum of 15 PSTs is reached. This support is within the constraints of the operating system or subsystem being used.

PWFI=

Specifies the pseudo-wait-for-input (PWFI) parameter. N is the default.

Y Activates pseudo wait-for-input (PWFI). If the application program issues a Get Unique (GU) call to get a message from the IMS message queue, MODE=SINGLE is specified on the TRANSACT macro and no message is available, IMS checks for other work for this region. If no other work is available, instead of returning status QC to the application program, IMS enqueues this region on scheduler subqueue 6 indicating that it is a PWFI region. If the next message is for the transaction scheduled in this MPP region, this region is dequeued from subqueue 6 and posted. IMS then returns the new message to the application program. This eliminates the rescheduling of IMS resources.

Related Reading: For more information on PWFI, see *IMS/ESA Administration Guide: System*.

N Disables PWFI. If the program issues a Get Unique (GU) call to get a message from the IMS message queue but no message is available, the application program receives status QC.

QBUF=

Specifies the one- to four-digit number of message queue buffers in subpool 0 to be allocated to the queue pool. In the shared queues environment, the number you specify for QBUF= is used as the initial number of message queue buffers allocated to the queue pool. The number of message queue buffers is dynamically expandable in the shared queues environment. The minimum number you can specify is three and the maximum is 9999. If you specify a value less than three, the number of buffers in the queue pool is set to three. If you do not specify a value, the number defaults to the value generated during sysgen for the BUFFERS= parameter of the MSGQUEUE macro.

QBUFHITH=

Specifies a one- to three-digit number from 1 to 100 that establishes the high

Parameter Descriptions

threshold percentage for the message queue buffer. When the buffer reaches the high threshold, it is dynamically expanded. If you do not specify a value for this parameter, the default is 80%.

QBUFLWTH=

Specifies a one- to three-digit number from 1 to 100 that establishes the low threshold percentage for the message queue buffer. When the message queue buffer reaches the low threshold, it is compressed. Compression only occurs when the number of message queue buffers exceeds the number allocated by system generation. If you do not specify a value for this parameter, the default is 50%. This parameter is valid only in the shared queues environment.

QBUFMAX=

Specifies the one- to four-digit number that indicates the maximum number of message queue buffers for the queue pool. The minimum number you can specify is 200 and the maximum is 9999. If you specify a value less than 200, the number of buffers in the queue pool is set to 200. If you do not specify a value for this parameter, the number of buffers in the queue pool is unlimited.

This parameter is only valid in a shared queues environment. If you do not specify a value for the SHAREDQ= parameter, this parameter is ignored.

QBUFPCTX

Specifies a one- to three-digit number in the range of 1 to 100 that establishes the percentage of the originally allocated message queue buffers that are dynamically expanded when the limit specified by the QBUFHITH= parameter is reached. The default for this parameter is 20%.

QBUFSZ=

Specifies a one- to five-digit number which represents the size in bytes of the message queue buffers (incore buffers) used by the queue manager. You must specify a size greater than or equal to the length specified by LGMSGSZ=. The size must be equal to or less than 30632 bytes which is the maximum length of the long message record. The size specified is rounded up to the nearest multiple of four. If you specify a value greater than 30632, the size is set to 30632. If you do not specify a value, or if you specify a value less than the length of the long message record, the size is obtained from the record length of the long message record. The DCBs for the message queue data sets are built during IMS sysgen processing, even though the data sets are not used.

This parameter is only valid in the shared queues environment. If you do not specify a value for the SHAREDQ= parameter, this parameter is ignored. If you do specify a value for the SHAREDQ= parameter, the value specified in the QBUFSZ= parameter is used to determine the size of the message queue buffers.

QTL=

Specifies the lower threshold percentage (1 to 99) for each message queue data set. The default is 60%. If this threshold is crossed and if the associated exit has not been modified, a message is issued.

QTU must be higher than QTL, and 0 is not valid. Either error results in the defaults being used.

Related Reading: For more information on message queue data sets, see *IMS/ESA Customization Guide*.

QTU=

Specifies the upper threshold percentage (2 through 100) for each message

Parameter Descriptions

queue data set. The default is 75%. If this threshold is crossed, and the associated exit routine has not been modified, a message is issued.

RCF=

Specifies whether RACF is to be used for transaction authorization or signon authorization checking.

A Includes options T, C, and S.

C Specifies that RACF is to be used for ETO terminal command authorization.

N Specifies that no signon, transaction, or command authorization is to be performed by RACF.

S Specifies that RACF is to be used for static and ETO terminal command authorization.

T Specifies that RACF is to be used for signon and transaction authorization.

Y Includes options T and C.

If your installation has more than 65536 terminals, you must use RACF for static LTERM security.

If this parameter is not specified, the default is the value specified at system definition.

RCFTCB=

Specifies a number between 1 and 20 indicating how many RCF TCBs are defined in your system. If you specify a null value or an invalid value, IMS uses the default of 1 RCF TCB. Increasing the number of RCF TCBs allows you to customize your system to achieve maximum parallelism and improved performance during RACF sign on and sign off calls.

RECA=

Specifies the number of receive-any buffers (1 to 500). This parameter overrides what is specified in the RECANY= parameter on the COMM macro.

RECASZ=

Specifies a one- to five-digit number indicating the size of the receive-any buffers. Valid values are from 112 to 30720. If you specify a null or invalid value, IMS uses the value specified in the RECANY parameter of the COMM macro used in the IMS generation. If that value is 0, IMS uses a default value of 2100.

RES=

Specifies whether (Y) or not (N) the PSBs or DMBs defined in system definition macros APPLCTN or DATABASE as RESIDENT should be made resident during system initialization. The default is Y.

REST=

Specifies a two-digit number. A number greater than 00 indicates that the utility is to be restarted. A specification of 00 indicates no restart.

REST=*nn* specifies from which set of utility control statements starts processing. Values 00 and 01 start with the first set. Sets of utility control statements with lower values than that specified by *nn* are not processed. A set of control statements is defined by the GO statement.

Parameter Descriptions

RGN=

Specifies the region size for this execution. For Fast Path, the default is 100K.

If TLIM is specified as a number greater than one, the RGN symbolic parameter must be increased approximately 50K bytes to allow virtual storage overhead for handling RE-IN-STATE situations. Insufficient virtual storage specification results in an S106 abend of the IFP region.

RSRMBR=

Specifies (for RSR systems only) the two-digit suffix for an RSR member DFSSRxx in IMS.PROCLIB.

RGSUF=

Specifies a three-character suffix for DFSPBxxx. This member contains default values for JCL EXEC statement parameters. It minimizes the number of EXEC parameter overrides needed so the EXEC parameter string does not exceed the 100 byte limit.

RST=

Specifies UCF restart. A value of 0 (no) or 1 (yes) must appear in the generated JCL statement for this parameter.

Related Reading: For more information on UCF restart, see *IMS/ESA Utilities Reference: Database Manager*.

RVFY=

Specifies if password reverification is activated (Y) or not (N). The default is N.

SAV=

Specifies a one- to three-digit number of dynamic save area sets for communication terminal I/O requests. IMS expands the number of SAPs up to ten times this number as it needs to process the work being given to the system.

You can use the formula below for calculating the number of SAPs your IMS system requires.

Formula for calculating the number of SAPs:

$$\begin{aligned} & ((\# \text{ of VTAM terminals} + 39) \div 40) \\ & + ((\# \text{ of MSC links} + 1) \div 2) \\ & + 1 \text{ (if VTAM required)} + 60 \text{ (if ETO defined)} \\ & + 4 \text{ (base itasks)} \end{aligned}$$

SCOPE=

Specifies whether intersystem sharing is to be performed. If SCOPE=LOCAL is specified, sharing is limited to intrasystem, and XCF and SLM are neither required nor used. If SCOPE=GLOBAL or SCOPE=NODISCON is specified, intersystem sharing is performed, and both XCF and SLM are required. This parameter must always be specified, because no default value exists.

With NODISCON, there is less impact on other systems when an IMS fails because MVS is not required to perform certain recovery actions when IRLM DISCONNECTS from the group. NODISCON can also mean that IMS restarts more quickly after an IMS terminates normally or abnormally because it does not have to wait for IRLM to rejoin the IRLM data sharing group. Under normal situations, IRLM Disconnects from the group ONLY when there are no IMS systems identified to it. NODISCON has no affect as long as there is at least one IMS identified to the IRLM.

Parameter Descriptions

Sharing can be performed between up to 248 IRLMs on a single MVS. This can be done for pre-installation testing. In this case, SCOPE=GLOBAL must be specified for all IRLMs.

SGN=

Specifies whether the signon verification security tables are to be loaded from IMS.MATRIX.

Related Reading: For more information, see “Establishing IMS Security” in *IMS/ESA Administration Guide: System*.

The possible values are:

- F** Specifies that the MTO cannot negate loading of the signon verification security tables from IMS.MATRIX.
- G** Includes options F and M.
- M** Specifies that a single user ID can sign on to multiple terminals. If the user structure name is different than the user ID for an ETO user (DFSSGNXO exit returned a user structure name that is not the same as the user ID), SGN=M must be specified for this user to be able to have multiple signons.
- N** Specifies that the signon verification security tables are not to be loaded unless overridden by the MTO. For a cold start, this implies that signon verification security is not going to be in effect.
- Y** Specifies that the signon verification security tables are to be loaded unless overridden by the MTO.
- Z** Includes options Y and M.

If all terminals are dynamic (required to sign on), or if you have no static terminals that are required to sign on, do not specify an SGN option other than M or N. However, when SGN=M is specified, specifying RCF=A or S, or specifying TRN=F or Y indicates that the signon verification tables are to be loaded. If you specify SGN=G or Z, either of the following messages might be issued erroneously:

- DFS171A with RETURN CODE=1000 (during restart)
- DFS3436A with RETURN CODE=0408 (during online change)

SHAREDQ=

Specifies a three-character suffix for the shared queues IMS.PROCLIB member, DFSSQxxx. When you specify this parameter, IMS uses the Common Queue Server (CQS) to place messages on the shared queues that reside in a coupling facility structure. If you do not specify this parameter, messages are placed on the message queue data sets.

There is no default for this parameter, it must be explicitly specified.

SHMSGSZ=

Specifies a one- to five-digit number that represents the length in bytes of a short message record. The length specified is rounded up to the nearest multiple of four. The length of the short message record cannot exceed 30632 bytes. The length must also be greater than or equal to the number of bytes in two times the maximum message prefix length plus four. If you specify a value greater than 30632, the size is set to 30632.

Parameter Descriptions

This parameter is only valid in the shared queues environment. If you do not specify a value for the SHAREDQ= parameter, the SHMSGSZ= parameter is ignored. If you specify a value for SHAREDQ=, but not for SHMSGSZ=, the size of the short message record is obtained from the DCB generated for the short message queue data set. The DCBs for the message queue data sets are built during IMS gen processing even though the data sets are not used.

Choose values for the SHMSGSZ= and the LGMSGSZ= based on the sizes of the messages processed by the queue manager. When determining which size message queue record to use, the queue manager calculates the size of the message prefix, multiplies the size by two and then adds the length of the user data. If the calculated value is less than or equal to the size of the short message queue buffer, the short message queue buffer is used. If the calculated value is greater than the size of the short message queue buffer, a long message queue buffer is used. The length of the message prefix varies based on the IMS system options specified.

Related Reading: For a list of message prefix lengths see Table 17 on page 118.

SOD=

Specifies a one-character class for SYSOUT to be used for the spinoff main storage dump. If omitted or specified as 0, no spinoff dump is taken.

SOUT=

Specifies the class assigned to SYSOUT DD statements. The default is A.

SPAP=

This parameter is no longer used.

SPIE=

Specifies the SPIE option to:

- 0** Allow your SPIE, (the SPIE established by the application program), if any, to remain in effect while the application program call is being processed.
- 1** Negate your SPIE while the application program call is being processed. Negated SPIEs are reinstated before returning to the application program.

Under MVS, if SPIE=1, a SPIE must be established; otherwise, system abend ABEND46D is issued.

A value of 0 or 1 must appear in the generated JCL statement for this parameter.

Related Reading: For further information about IMS and SPIEs, refer to the topic, "How PL/I-IMS Error Handling Operates" in *OS PL/I Version 2 Programming Guide, SC26-4307*.

SPM=

Specifies a two-character suffix for the storage pool manager PROCLIB member, DFSSPMxx.

SRCH=

Is the module search indicator for directed load.

- 0** Standard search
- 1** Search JPA and LPA before searching PDS

SSM=

Specifies a one- to four-character identifier. When building IEBUPDTE JCL, you must generate the member name by concatenating this SSM identifier to the IMSID.

All external subsystems to be accessed by any region must be defined in this member. Essentially, the member specified by the IMS procedure is a master list. Dependent region procedures (DFSMPR, IMSBATCH, and IMSFP) can define subset lists. Each subset list can define all, some, or none of the external subsystems defined in the IMS master list. The SSM entry in each dependent region controls that region's access to subsystems.

Specifying SSM in a dependent region procedure but not in the IMS procedure is invalid, and no connection is made.

- To allow a region to access all subsystems defined in the IMS procedure, do **not** code SSM. However, if preferred, the SSM entry can specify the same member as the IMS procedure.
- To allow access to selected subsystems, specify a member defining only those subsystems a region is allowed to access. For example, if the IMS procedure member defines subsystems A, B, and C, but a region is only to access A and C, the region's SSM should specify a member defining only A and C. Thus, the SSM member is a subset of the IMS procedure member.
- To prevent this region from accessing any subsystem, specify a member containing no entries.

STIMER=

Restriction: The following description applies to online systems only.

Specifies the processor time statistics to be gathered.

- 0** No processor time statistics are to be gathered. No STIMER/TTIMER sequence is issued.
- 1** No DL/I processor time is to be included in the processor time statistics. An STIMER/TTIMER sequence is issued once for each program invocation and once for each DL/I call.
- 2** If the LSO=S region startup parameter has been selected, processor time statistics include both application program time and DL/I processing time. One STIMER/TTIMER sequence is issued for each program invocation. This is the default. This value allows the maximum amount of time needed for transaction PROCLIM timing to be effective.

If you use the IMS timing services STIMER=1 or STIMER=2, you should ensure that your application programs do not use or invoke SVC2E or SVC2F (STIMER or TTIMER). If these services are invoked, they negate the function of controlling application programming loops, and invalid time statistics might be gathered. Additionally, ABENDU240 might occur. You should also be aware that the LSO option you select affects the meaning of the processor time statistics. If LSO= is specified, most DL/I processing time is not included in the statistics.

STIMER=

Restriction: The following description applies to batch systems only. Specifies whether (1) or not (0) the timer is to be set.

If STIMER=0 and a value is not entered for the CPUTIME= parameter, the time is not set. If STIMER=0 and a value *is* entered for the CPUTIME= parameter,

Parameter Descriptions

the STIMER= parameter defaults to 1 and the timer is set. If nothing is entered for either the STIMER= or CPUTIME= parameters, the STIMER= parameter defaults to 0 and the timer is not set.

If STIMER=1, the timer is set and processor time statistics are gathered. The statistics include both application program time and DL/I processing time if the LSO=S option has been specified; most DL/I processing time is not included if LSO=Y is specified. One STIMER/TTIMER sequence is issued for each program invocation.

STIMER=1 should only be specified for a BMP when gathering statistics. Specifying STIMER=1 for a BMP is similar to specifying STIMER=2 for an MPP.

SUF=

Specifies the one-character suffix for the control program name. This allows multiple copies of the IMS nucleus to reside on IMS.RESLIB.

This parameter cannot be changed at emergency restart.

SWAP=

Makes address space swappable (Y) or nonswappable (N). The default is Y.

If you are using DBRC, you might want to choose option N because of the possibility of being swapped out while holding the reserve on RECON.

If you are using IRLM for block-level data sharing, the SWAP parameter has no effect. If you specify IRLM=Y, a batch job is always made non-swappable to prevent the job from obtaining locks for resources and then being swapped out.

SYS=

Specifies an optional second-level dsname qualifier for those data sets designated as "mandatory shared" in an XRF complex. When specified, the operand must be enclosed in single quotes and must include a trailing period.

Example: `SYS='IMSA.'`

SYS1=

Specifies an optional second level dsname qualifier for those data sets designated as "mandatory replicate" in an XRF complex. When specified, the operand must be enclosed in single quotes and must include a trailing period.

Example: `SYS1='IMSA.'`

SYS2=

Specifies an optional second-level dsname qualifier for those data sets designated as "optional replicate" in an XRF complex. When specified, the operand must be enclosed in single quotes and must include a trailing period.

Example: `SYS2='IMSA.'`

TEST=

Specifies whether (1) or not (0) the addresses in your call list should be checked for validity. A value of 0 or 1 must appear in the generated JCL statement for this parameter. An address is invalid if it is either lower than the lowest address not in the MVS nucleus or higher than the highest address in virtual storage of the machine.

TLIM=

Specifies a two-digit termination limit option with a decimal number between 01

Parameter Descriptions

and 99. When the number of application program abends reaches this limit, the message region is automatically terminated. The accumulated number of abends for a region is incremented only if the transaction is not requeued. TLIM is not incremented in the case of a pseudoabend. This allows MVS to print the accumulated SYSOUT data sets. If omitted or if 0 is specified, no limit on program abends is set, and the message region is not automatically terminated. For an application defined as PGMTYPE=BATCH in the APPLCTN macro statement, the optional parameter is ignored if specified. The default is 1.

TMINAME=

Specifies the transport manager instance name the batch job is to use. If TMINAME= is not specified in the DBBATCH procedure, the default TMI name is the TMI name specified in the IMSCTRL macro, or blanks if no TMI name is specified.

TRACE=

Specifies whether the IRLM is to turn on traces. Tracing is initialized at IRLM startup. The default is TRACE=NO.

NO Traces are not captured unless you use the MVS TRACE CT command.

YES Initiates IRLM tracing.

TRACK=

Specifies the level of recovery tracking for the RSR tracking subsystem.

TRACK=NO is used by the IMS active subsystem and XRF alternates subsystems, and specifies that no recovery tracking be done. This is the default, so existing procedures do not need to be changed.

TRACK=RLT is used by the RSR tracking subsystem, and specifies that recovery-level tracking be done.

TRACK=DLT is used by the RSR tracking subsystem, and specifies that database-level tracking be done.

If you specified RSRFEAT=RLT in the IMSCTRL macro, then TRACK can only be NO or RLT. If you specify TRACK=DLT, it will be changed to TRACK=RLT.

TRN=

Specifies whether transaction authorization checking is to be performed.

F Specifies that the MTO cannot negate transaction authorization checking when issuing the /NRESTART command.

Y Specifies that the MTO can optionally override transaction authorization checking by specifically requesting no transaction authorization checking on the /NRESTART command.

N Specifies that the MTO can optionally enable transaction authorization checking by specifically requesting transaction authorization checking on the /NRESTART command.

TSR=

Specifies the chosen time stamp representation.

U The time stamp extension to the I/O PCB contains UTC.

L The time stamp extension to the I/O PCB contains local time. The default is L extension to the I/O PCB. If you specify L, local time becomes the date extension to the I/O PCB.

Parameter Descriptions

The default is L.

UHASH=

Specifies the one- to eight-character name of the Fast Path user-hashing module pointed to by the USERLIB or IMS.RESLIB DD statement

In a multiple-IMS configuration, the same hashing module must be used for all systems. If you do not specify this parameter, the hash module specified on the USRHASH= parameter of the FPCTRL macro in system definition is used.

Related Reading: For more information related to HASH modules see *IMS/ESA Customization Guide*.

This parameter cannot be changed at emergency restart.

UHTS=

Specifies a one- to five-digit number of user (SPQB) hash table slots. Valid values are from 0 to 32767. The default is 256. If you specify a null or invalid value, 256 is assumed. For more information see “Specifying the Number of Hash Table Slots” on page 409.

USERVAR=

Specifies the user name for the IMS active subsystem. It is optional.

The *username* specified for the USERVAR= parameter for an RSR active site must remain the same in the event of a takeover and restart from a remote site. Likewise, the logical unit type 6.2 (LU 6.2) *uservars* specified at an RSR active site must remain the same in the event of a takeover and restart from a remote site.

For an XRF-capable system, the USERVAR= parameter will override the USERVAR specified in the DFSHSBxx procedure library member.

VALCK=

Specifies the validity-check option:

- 0** Don't check the validity of addresses in the user's call list. This is the default.
- 1** Check the validity of the addresses in the user's call list.

An address is invalid if it is either lower than the lowest address not in the MVS nucleus or higher than the highest address in virtual storage.

VAUT=

Specifies whether (1) or not (0) IMS is to use the VTAM authorized path facility.

VFREE=

Specifies whether (Y) or not (N) the virtual storage obtained by virtual fetch to hold application programs is to be freed when the programs issue RETURNS. The default is N (storage not freed).

VSFX=

A two-digit suffix for the PROCLIB member with the names of the non-preloaded application programs for which IMS is to have virtual fetch management.

VSPEC=

Specifies a two-character suffix for DFSVSMxx, the IMS.PROCLIB member that contains control statements. The default is 00. The default member in IMS.PROCLIB is DFSVSM00.

Control statements in DFSVSMxx define the following:

- Size and number of buffers in a subpool
- Number and use of local shared resource pools
- Whether sequential buffering is to be used
- Various performance and trace options
- DASD logging data set requirements
- Whether dynamic allocation for IMS batch is to be used
- Coupling facility structure names for sysplex data sharing

The VSPEC= parameter performs the same function in an IMS online system that the DFSVSAMP DD statement performs in a batch system. For more information, see “Member DFSVSMxx” on page 339.

WADS=

Specifies whether single (S) or dual (D) logging is to be done on the write ahead data set. The default is S.

This parameter cannot be changed at emergency restart.

WKAP=

Specifies the amount of subpool 231 storage to be allocated to the work area pool. The value can be specified as one- to six-numeric characters or one- to five-numeric characters followed by K (kilobyte), M (megabyte), or G (gigabyte). If K, M, or G is not specified, K is the default. The maximum value that can be specified is 2G-1. If the value specified exceeds 2G-1 the default is 2G-1. The value entered is rounded up to the nearest 4K page boundary. The default is 5K. The WKAP pool size cannot be specified at system definition.

x, y, and z

Specify the library, volume, and unit type values. The library must be an MVS-authorized library.

YEAR4=

Specifies whether the year is displayed as four digits in the time stamp as a result of the display command. The default is YEAR=N.

N Displays two digits.

Y Displays four digits.

DD Statement Descriptions

The following list contains a description of all the DD statements that can be used by the procedures described in this chapter.

DFSCCTL DD

Contains the SETR or SETO statements to specify the processing options and ranges required at scheduling time. The SETO (SET Options) statement provides the ability to set the processing options to control the execution of HSSP. The SETR (SET Range) statement restricts access to specified parts of a DEDB.

DFSDUPPM DD

Is generated when a second address is specified for a primary BTAM master terminal on the system definition LINE macro. The second address is for the primary BTAM master terminal on the XRF alternate system. A DD statement named DFSDUPSM can be generated for an XRF alternate secondary master terminal.

DD Statement Descriptions

DFSESL DD

When the SSM parameter is specified for the IMS control region, you must load certain subsystem modules from subsystem libraries. These libraries, which must be APF authorized, must be available to IMS. If JOBLIB/STEPLIB/LINKLIST concatenation is not authorized, you must use the DFSESL DD statement. For online IMS regions, the subsystem library or libraries must be concatenated after the library containing the IMS modules (usually IMS.RESLIB). When multiple subsystems are connected, additional subsystem data sets can be concatenated. An example follows:

Example:

```
//DFSESL DD DISP=SHR,DSN=IMS.RESLIB
//          DD DISP=SHR,DSN=DSNxxx.DSNLOAD
//          DD DISP=SHR,DSN=DSNyyy.DSNLOAD
```

IMS.RESLIB is first, followed by the subsystem libraries.

DFSOLPnn DD

Defines the primary online DASD log data set, where *nn* can be any numeric value. You can specify between 3 and 100 primary DASD data sets. The block size for the online log data sets must be a multiple of 2K bytes.

Related Reading: For more information, see “Allocate and Catalog IMS System Data Sets” in *IMS/ESA Installation Volume 1: Installation and Verification*.

DFSOLSnn DD

Defines the optional secondary online DASD log data sets, where *nn* can be any numeric value. You can specify between 3 and 100 secondary DASD data sets. The block size for the online log data sets must be a multiple of 2 K. These statements are required only if dual logging is requested. Suffixes must be consistent with the primary log data sets.

DFSRESLB DD

Points to an authorized library that contains the IMS SVC modules. For IMS batch, IMS.RESLIB and any data set that is concatenated to it on the DFSRESLB DD statement must be authorized through the Authorized Program Facility (APF). For more information, see “Specifying the DFSRESLB DD Statement in IMS Batch Procedures” on page 192.

DFSSTAT DD

Defines a data set describing DB call and buffering activity during an application’s execution. The reports are written when the application terminates. If you are interested in receiving //DFSSTAT reports, include a //DFSSTAT DD statement in this procedure.

Example: //DFSSTAT DD SYSOUT=A

Related Reading: For more information on //DFSSTAT reports, see *IMS/ESA Administration Guide: Database Manager*.

*DFSTCF DD

Defines the data set specified if the time-controlled operation function is used. The asterisk must be deleted to activate the time-controlled operations function.

DFSVSAMP DD

Defines the following:

- Size and number of buffers in a subpool
- Number and use of local shared resource pools

- Whether sequential buffering is to be used
- Various performance and trace options
- DASD logging data set requirements
- Whether dynamic allocation for IMS batch is to be used
- Coupling facility structure names for sysplex data sharing
- Caching option for the OSAM subpools.

Related Reading: For additional information on OSAM subpools, see “OSAM Subpool Definition” on page 379.

The DFSVSAMP DD statement performs the same function in an IMS batch system that the DFSVSMxx DD statement performs in an online system. In a batch system, the control statements defining the preceding options are in a data set with the ddname of DFSVSAMP; in an online system, the control statements are in a member of the IMS.PROCLIB data set with the member name of DFSVSMxx.

For more information on these control statements, see “Member DFSVSMxx” on page 339.

DFSWADSn DD

Defines the required write-ahead data sets, where *n* can be a number from zero to nine. If dual logging to the WADS is requested, at least two WADS data sets must be provided. If an I/O error occurs on the current WADS, the next WADS is used.

Dual WADS logging provides backup on a read error while terminating the OLDS from the WADS. The primary and secondary WADS contain the same data. Single or dual WADS logging is determined from an execution-time parameter you specify. Regardless of whether dual logging is selected, as many as ten WADS DD statements can be included. The extra data sets serve as spares on write errors. When a write error occurs, write-ahead logging continues whether single mode or dual mode is selected. One of the spare data sets is substituted for the one with the error. When required WADSs are unavailable because of I/O errors, IMS issues a warning message and continues processing in a degraded mode, truncating OLDS buffers for LWA. When switching to a new WADS after a write error, the current online log buffer is truncated and written to the OLDS, thereby making obsolete all data currently in the WADS.

FORMATA DD and FORMATB DD

Points to IMS.FORMAT A and IMS.FORMAT B, which contain online MFS definitions to be used as the format library by the online system. They are required by MFS-supported terminals.

IMS limits the number of data sets that can be concatenated for MFS format libraries, IMS.FORMAT A and IMS.FORMAT B, to 16. All the concatenations must have like attributes.

FPTRACE DD

Defines the destination for the Fast Path trace output. Activate the trace with the DD statement in the procedure and the following command:

```
/TRACE SET ON TABLE FAST
```

IEFRDER DD

Defines the primary system log data sets. If DASD instead of tape is used for logging, substitute the appropriate DD statements. IEFRDER and IEFRDER2

DD Statement Descriptions

can be a combination of tape and DASD. This statement is not required if the job does not declare database-update intent.

IEFRDER2 DD

Defines the secondary system log data sets. This statement is included only when dual system log data sets are used. If DASD is used for logging instead of tape, substitute the appropriate DD statements. IEFRDER and IEFRDER2 can be a combination of tape and DASD.

IMS DD

Add an IMS DD statement for IMS.PSBLIB, concatenated with IMS.DBDLIB, if GSAM or GLOBAL databases are accessed by the batch application. The statements are:

```
//IMS DD DSN=IMS.PSBLIB,DISP=SHR
// DD DSN=IMS.DBDLIB,DISP=SHR
```

The PSB for the batch application program must be contained in the IMS.PSBLIB, and the DBDs for the GSAM or GLOBAL databases referenced by the PSB must be contained in the IMS.DBDLIB.

IMSACBA DD and IMSACBB DD

Point to partitioned data sets containing the output from ACBGEN. Two statements are needed for online change. If any DOPT PSBs exist for an online system, they must reside in any data set in any concatenation other than the first. If MODSTAT is omitted, IMSACB DD is used.

IMSIRD DD

Defines the data set that submits the modified start-up JCL for a dependent region to the internal reader.

IMSLOGR

Defines the input log data set for extended restart. This statement is required if an application program is performing an XRST call. Any program that issues symbolic checkpoint calls must also issue the XRST call and therefore requires the IMSLOGR DD statement.

IMSMON DD

Describes the recording device that the IMS Monitor uses. This statement is produced only if MON=Y is specified in the IMSCTF macro.

IMSRDS DD

Defines the first restart data set; this data set contains information required for recovery, including the checkpoint ID table needed for restarting IMS. This data set does not contain any log records.

IMSRDS2 DD

Defines the second restart data set. It is used only in an XRF system. This data set is identical in function to IMSRDS DD.

IMSTFMTA DD and IMSTFMTB DD

Point to IMS.TFORMAT, which contains the online MFS descriptors for test mode online execution. These DD statements consist of IMS.TFORMAT, concatenated in front of IMS.FORMATA and IMS.TFORMAT, concatenated in front of IMS.FORMATB. If you change MFS formats online, the two DD statements might point to this single TFORMAT data set, or the DD statements might point to two separate TFORMAT data sets.

JCLOUT DD

Defines a data set where output produced by a GENJCL command is to be written.

JCLPDS DD

Defines a partitioned data set that contains the skeletal JCL execution and default members used by the GENJCL commands.

LGMSG DD

Points to the normal long message data set. If a batch only execution is planned, you do not need the message queue data sets. The Queue Manager Concurrent I/O provides a facility for the IMS customer to provide multiple "Normal" short and long message queue data sets. This facility is optional and is invoked by the IMS customer providing from one to ten DD statements for the "Normal" short and long message queues data sets.

The current implementation of the "Normal" short and long message queues allows only one DD statement for each.

LGMSGL DD

Points to the local long message data set. This statement is used only in an XRF complex.

MATRIXA DD

Points to IMS.MATRIXA, which contains the security tables created by the SMU. With this DD statement, you can bring the tables online via a /MODIFY command when IMS.MATRIXB is actively being used by the control region. At any given time, the active MATRIX data set (in use in the online system) corresponds directly to the active MODBLKS set. (They must have the same suffix.) If MODBLKSA is the active data set, security data is taken from MATRIXA. It must be authorized by APF.

Related Reading: For more information on IMS.MATRIXA, see *IMS/ESA Administration Guide: System*.

MATRIXB DD

Points to IMS.MATRIXB, which contains the security tables created by the SMU. With this DD statement, you can bring the tables online via a /MODIFY command when IMS.MATRIXA is actively being used by the control region. At any given time, the active MATRIX data set (in use in the online system) corresponds directly to the active MODBLKS set. (They must have the same suffix.) If MODBLKSB is the active data set, security data is taken from MATRIXB. It must be authorized by APF.

Related Reading: For more information on IMS.MATRIXB, see *IMS/ESA Administration Guide: System*.

MODBLKSA DD

Points to IMS.MODBLKSA, which contains the system definition output to be brought online via a /MODIFY command when IMS.MODBLKSB is actively being used by the control region. MODBLKSA must be authorized by APF.

MODBLKSB DD

Points to IMS.MODBLKSB, which contains the system definition output to be brought online via a /MODIFY command when IMS.MODBLKSA is actively being used by the control region. MODSBLKB must be authorized by APF.

MODSTAT DD

Points to IMS.MODSTAT, which indicates which of the following active or inactive data sets the IMS online system should use during initialization:

- ACBLIBA or ACBLIBB
- FORMATA or FORMATB
- MODBLKSA and MATRIXA or MODBLKSA and MATRIXB

DD Statement Descriptions

When an online system is using online change, the MODSTAT data set keeps track of which libraries are active. When a batch system uses an ACBLIB data set that is a DBBATCH procedure, the MODSTAT data set determines which set of libraries is to be used.

The presence of the MODSTAT DD statement indicates that IMS.MODSTAT determines which ddname (IMSACBA or IMSACBB) to use. If MODSTAT is omitted, IMSACB DD is used.

MODSTAT2 DD

Points to IMS.MODSTAT2, which indicates the active or inactive data sets that the IMS online system should use during initialization. This statement is used only in an XRF system.

MSDBCP1 DD and MSDBCP2 DD

Defines the first pair of MSDB checkpoint data sets. IMS alternates between the two when taking an MSDB checkpoint. During an /NRE, /ERE, or IMS procedure, the most current MSDB is determined, and the MSDBs are loaded from that data set (unless the MSDBLOAD keyword is specified during warmstart).

MSDBCP3 DD and MSDBCP4 DD

Defines the second pair of MSDB checkpoint data sets. Used only in an XRF environment, these data sets are identical in function to MSDBCP1 and MSDBCP2. In an XRF environment, you must specify all four data sets, because both the active and the alternate subsystem each require two. Any two of the four can contain the latest MSDB checkpoint. Although an active subsystem can select the data set containing the latest MSDB checkpoint, the alternate subsystem must select the two data sets not used by the active.

MSDBDUMP DD

Defines a data set that contains a dump of all MSDBs. This data set specifies where to dump the MSDBs when a /DBD DB MSDB command is entered. Successive executions of the command cause the previous contents to be overlaid.

MSDBINIT DD

Defines a data set that contains unloaded or reconstructed MSDBs. This data set specifies where to start the initial load of the MSDBs during all cold starts and during a normal restart if the MSDBLOAD parameter is specified for the /NRESTART command. It is produced by executing the MSDB Dump Recovery or MSDB Maintenance utility. MSDBINIT can contain one, several, or all MSDBs defined.

PRINTDD DD

Defines the output data set for the test program, including displays of control blocks using the SNAP call. It must conform to the MVS SNAP data set requirement.

PROCLIB DD

Points to IMS.PROCLIB, which contains all IMS-generated cataloged procedures, jobs, and control statements.

QBLKS DD

Points to the normal queue blocks message data set. If a batch-only execution is planned, you do not need the message queue data sets.

QBLKSL DD

Points to the local queue blocks message data set. This statement is used only in an XRF complex.

RECON n DD

Defines the DBRC RECON data sets, where n is 1, 2, or 3. Add RECON DD statements if DBRC is being used. If dynamic allocation is being used, the DD statements are not necessary. The names of these data sets must be consistent with all subsystems that share the RECON data set.

Related Reading: For more information on creating a RECON data set, see *IMS/ESA DBRC Guide and Reference*.

RESLIB DD

Defines the data set that contains the IMS nucleus and required action modules. Must be authorized by APF.

SHMSG DD

Points to the normal short message data set. If a batch-only execution is planned, you do not need the message queue data sets.

SHMSGL DD

Points to the local short message data set. This statement is used only in an XRF complex.

STEPLIB DD

Points to IMS.RESLIB, which contains the IMS nucleus and required action modules. If STEPLIB is unauthorized because of having unauthorized libraries concatenated to IMS.RESLIB, a DFSRESLB DD statement must be included. The STEPLIB statement need not be authorized for IMS batch.

SYSABEND DD

Defines a dump data set. If both a SYSABEND DD statement and a SYSUDUMP DD statement are used, the last occurrence is used to define the dump.

SYSMOD DD

Defines the destination of the output for the linkage editor.

SYSPRINT DD

Defines the output data set for control messages, statistics and reports produced by a utility.

SYSTSPRT DD

Defines the output data set for TSO Parse. The data set contains all TSO Parse error messages, such as errors in parsing the DFSRSRxx PROCLIB member. This DD statement is optional and is not generated by IMS.

SYSUDUMP DD

Defines a dump data set. If both a SYSABEND DD statement and a SYSUDUMP DD statement are used, the last occurrence is used to define the dump.

SYSIN DD

Defines the control statement input data set.

SYSLIB DD

Points to the libraries that contain the other modules needed for the final executable module.

SYSLIN DD

Defines the destination of the output from the compiler.

SYSUT n DD

Is a temporary work data set used by the compiler, where n is 1, 2, 3, or 4. This data set must reside on DASD.

DD Statement Descriptions

Chapter 5. IMS Dumping and Dump Formatting Options

In this Chapter:

- “Installing the Interactive Dump Formatter”.
- “Dumping Options”.
- “IRLM Use of SDUMP” on page 293.

Installing the Interactive Dump Formatter

Prerequisite: Before you begin installing the Interactive Dump Formatter, be sure you have IPCS already functioning with ISPF/PDF.

1. Update the following DD concatenations:

DDNAME	Data Set to Be Added	Contents of Data Set
SYSPROC	IMS.DFSCLST	CLISTS
ISPMLIB	IMS.DFSMLIB	Messages
ISPLLIB	IMS.DFSPLIB	Panels
ISPTLIB	IMS.DFSTLIB	Tables
IPCSPARM	IMS.MACLIB	All IMS macros Note: DFSIPCSP is the only member that is used from this data set.
TASKLIB	IMS.RESLIB	Formatting modules Note: The TASKLIB concatenation is specified as part of the IPCS command invocation.

2. Start IPCS and select the dump you wish to browse.
3. From the IPCS Primary Option Menu, select Option 2.6, ANALYSIS.COMPONENT.
4. From the Component Analysis Panel, select DFSAAMPR.
5. From the IMS Dump Formatting Primary Option Menu, select Option 0, INIT.
6. From the IMS Dump Contents Status Panel, enter the job name of either the IMS control/batch region or the IMS DLISAS region.
7. Press the Enter key and then PF3 to return to the IMS Dump Formatting Primary Option Menu.
8. Select an IMS Analysis Option from the Primary Option Menu. HILEVEL is a good starting point because it provides a broad diagnostic overview of the problem.

Dumping Options

The dumping options available to you include the following:

- Offline IMS or MVS formatting from a SYS1.DUMPxx (SDUMP) data set
- Offline IMS or MVS formatting from a SYSMDUMP data set
- Online IMS or MVS formatting directed to either a SYSABEND or SYSUDUMP SYSOUT class
- Online MVS formatting directed to either a SYSABEND or SYSUDUMP spinoff dump SYSOUT class

Dumping Options

The dumping options in effect in an IMS environment depend on:

- Whether execution is online or batch.
- Whether the error might terminate IMS.
- The FMTO EXEC parameter specification.
- The SOD EXEC parameter specification (online only).
- The choice of a SYSABEND, SYSUDUMP, or SYSMDUMP DD statement.
- The MVS dump options chosen for SDUMP, SYSABEND, SYSUDUMP, and SYSMDUMP.
- Whether the Dump Override Table contains any entries.

Related Reading: See *IMS/ESA Customization Guide* for further information about Dump Override Tables.

MVS Dump Options

In order for IMS dumps to be useful in diagnosing problems, the proper dump options must be used (by specification or default) for each type of MVS dump used for IMS.

The MVS operator command DISPLAY DUMP, OPTIONS can be used to determine the dump options currently in effect on your system.

If the current dump options do not include the options listed below, the defaults obtained from SYS1.PARMLIB (members IEAABD00, IEADMP00, and IEADMR00) must be altered. The MVS operator command CHNGDUMP can be used to change dump options.

- SDUMP options
SDUMP=(CSA,LSQA,PSA,RGN,SQA,SUM,SWA,TRT)
- SYSABEND options
SYSABEND,SDATA=(CB,DM,ENQ,ERR,IO,LSQA,SUM,TRT)
SYSABEND,PDATA=(JPA,LPA,PSW,REGS,SA,SPLS)
- SYSUDUMP options
SYSUDUMP,SDATA=(CB,ERR,SUM)
SYSUDUMP,PDATA=(JPA,LPA,PSW,REGS,SA,SPLS)
- SYSMDUMP options (**ONLINE** system only)
SYSMDUMP=(CSA,LSQA,RGN,SQA,SUM,SWA,TRT)
- SYSMDUMP options (**BATCH** system only)
SYSMDUMP=(LSQA,RGN,SQA,SUM,SWA,TRT)

Online System Dumps

In the online IMS environment, the following dumps are possible:

- SDUMP to a SYS1.DUMPxx data set. (SDUMP can then be formatted using the Offline Dump Formatter utility.)
This dump can be generated both for errors that cause IMS to terminate (*terminating errors*) and for errors that do not cause IMS to terminate (*nonterminating errors*).
- SYSABEND or SYSUDUMP (with IMS or MVS formatting performed online).
This dump is generated for terminating errors only if SDUMP fails or was not requested.
- SYSMDUMP to a data set. (SYSMDUMP can then be formatted using the Offline Dump Formatting utility.)

Dumping Options

This dump is generated for terminating errors only if SDUMP fails or was not requested.

- Spinoff dump to SYSABEND or SYSUDUMP (with MVS formatting performed).
This dump is generated for nonterminating errors and can be generated in addition to SDUMP.

The IMS dumping options are controlled by the Dump Override Table entries, the FMTO EXEC parameter specification for the IMS control region, and the presence of a SYSABEND, SYSUDUMP, or SYSMDUMP DD statement. (These DD statements are mutually exclusive; provide only one.)

Related Reading: The Dump Override Table (DFSFDOT0) and its relationship with the various dumps are explained in the *IMS/ESA Customization Guide*.

The FMTO options and their effect on the dumps produced are summarized in Figure 6.

FMTO Value	IMS Offline Formatting Sysudump		IMS Inline Formatting SYSABEND SYSUDUMP			MVS Online Formatting SYSABEND SYSUDUMP		IMS Offline Formatting SYSMDUMP		IMS Spinall Dump SYSABEND SYSUDUMP	
	TE	NT	TE	NT	FDDL	TE	NT	TE	NT	TE	NT
D	Y	Y	S	—	S	S	—	S	—	—	Y
X	Y	N	S	—	S	S	—	S	—	—	Y
M	Y	Y	N	—	N	S	—	S	—	—	Y
R	Y	N	N	—	N	S	—	S	—	—	Y
T	N	Y	Y	—	Y	Y	—	Y	—	—	Y
P	N	Y	Y	—	N	Y	—	Y	—	—	Y
F	N	N	Y	—	N	Y	—	Y	—	—	Y
N	N	Y	N	—	N	Y	—	Y	—	—	Y
Z	N	N	N	—	N	Y	—	Y	—	—	Y

TE Error would terminate IMS.

NT Error would **not** terminate IMS.

FDDL Formatted dump delete list processing.

Y Yes

S Yes, but only if SDUMP fails.

N No

— Not applicable, or not attempted.

Notes:

- The default is FMTO=D.
- SYSABEND, SYSUDUMP, and SYSMDUMP are mutually exclusive.
- MVS formatting always accompanies IMS online formatting.
- Spinoff dumps can occur in addition to SDUMPs.

Figure 6. Online FMTO Options

Dumping Options

When using this figure, it is important to remember the following:

- If SDUMP is requested by the FMTO option, it is attempted first.
If SDUMP is successful, no other dump is generated. (Spinoff dumps are an exception.)
- If SDUMP fails, if IMS online formatting is requested, and if a SYSABEND or SYSUDUMP DD statement is present, a dump with IMS and MVS online formatting is generated for terminating errors.
The formatted dump delete list (FDDL) consists of IMS module and control block names, and a dump or delete action indicator. For a module, the delete process removes storage occupied by the module. This causes the module to be omitted from the storage image portion of the dump. However, the module/save ID is printed in the dump. For a control block, the delete process similarly removes storage and storage image output if the block was successfully formatted.
- If SDUMP fails, IMS online formatting is not requested, and a SYSABEND or SYSUDUMP DD statement is present, a dump with MVS online formatting is generated for terminating errors.
- If SDUMP fails and a SYSDUMP DD statement is present, a machine readable dump is generated for terminating errors.
- If a spinoff dump SYSOUT class is specified (SOD EXEC parameter in the IMS control region), a spinoff dump is generated for nonterminating errors.

If SYSDUMP is used, some additional considerations are necessary:

- SYSDUMP DD statements should be present in the IMS, DBRC, and DLISAS procedures.
- The SYSDUMP DD statement must specify DISP=MOD, because it is possible for multiple tasks to generate dumps during the processing of an error.
- The SYSDUMP data set must be refreshed (for example, scratched and reallocated) before it can be reused. If this is not done, new dumps are appended to the end of the data set and may not be accessible to the Offline Dump Formatting utility.

Following are some additional considerations if spinoff dumps are used:

- Spinoff dumps allow the dump data set to be released to JES for immediate printing. (This can be useful when external subsystems are supported.)
- Spinoff dumps are only generated for those tasks that do not terminate IMS.
- If a SYSABEND or SYSUDUMP DD statement is not provided, the first dump is not printed.

For subsequent dumps, the dump data set is dynamically allocated using the spinoff dump class.

- The spinoff dump class (SOD execute parameter) should be the same as the SYSABEND or SYSUDUMP SYSOUT class.

If the classes are different, the first dump goes to the class specified on the SYSABEND or SYSUDUMP DD statement. The dump data set is then dynamically deallocated to free the dump for printing and then dynamically reallocated using the spinoff dump class. Therefore, all subsequent dumps go to the spinoff dump class.

- The only valid SYSABEND or SYSUDUMP DD statement parameter used for spinoff dump is the SYSOUT class. Other parameters are ignored.
- Spinoff dumps use the default output limit established during JES generation. Incomplete spinoff dumps can result if the limit specified is too low.
- SYSDUMP should not be used in place of SYSABEND or SYSUDUMP.

Online System Dependent Region Dumps

In the online IMS environment, the following dumps are useful for MPP, IFP, and BMP dependent regions:

- SYSABEND or SYSUDUMP (with MVS formatting performed online).
- Spinoff dump to SYSABEND or SYSUDUMP (with MVS formatting). This dump is available only for MPP and IFP dependent regions.
- SDUMP to a SYS1.DUMPxx data set. (SDUMP can then be formatted using the Offline Dump Formatter utility.) An SDUMP is produced in the dependent region for certain IMS system errors involving DL/I or Fast Path. In this instance, the SDUMP is controlled by the FMTO specification for the control region.

Except for the SDUMPs, the dumping options for the dependent regions are controlled by the following:

- The spinoff dump SYSOUT class (SOD EXEC parameter in the DFSMPR and IMSFP procedures)
- The presence or absence of a SYSABEND or SYSUDUMP DD statement in the dependent region JCL. (These DD statements are mutually exclusive; provide only one.)

SYSMDUMP should not be used in IMS dependent regions.

If spinoff dumps are used, some additional considerations are necessary:

- Spinoff dumps allow the dump data set to be released to JES for immediate printing. (This can be useful for “never-ending” MPP or IFP regions.)
- If a SYSABEND or SYSUDUMP DD statement is not provided, the first dump is not be printed. For subsequent dumps the dump data set is dynamically allocated using the spinoff dump class.
- The spinoff dump class (SOD EXEC parameter) should be the same as the SYSABEND or SYSUDUMP SYSOUT class.

If the classes are different, the first dump goes to the class specified on the SYSABEND or SYSUDUMP DD statement. The dump data set is then dynamically deallocated to free the dump for printing and then dynamically reallocated using the spinoff dump class. Therefore, all subsequent dumps go to the spinoff dump class.

- The only valid SYSABEND or SYSUDUMP DD statement parameter used for a spinoff dump is the SYSOUT class. Other parameters are ignored after the first dump.
- Spinoff dumps use the default output limit established during JES generation. Incomplete spinoff dumps can result if the limit specified is too low.
- If the COBOL DEBUG option is used, dumps cannot be generated.

Batch System Dumps

In the batch IMS environment, the following dumps are possible:

- SYSMDUMP to a data set. (SYSMDUMP can then be formatted using the Offline Dump Formatting utility.) This dump is only generated for terminating errors.
- SYSABEND or SYSUDUMP (with IMS or MVS formatting performed online). This dump is only generated for terminating errors.

Dumping Options

The IMS dumping options are controlled by the Dump Override Table entries, the FMTO EXEC parameter specification for the IMS batch region, and the presence of a SYSABEND, SYSUDUMP, or SYSMDUMP DD statement. (These DD statements are mutually exclusive; provide only one.)

Related Reading: The Dump Override Table (DFSFDOTO) and how its relationship with the various dumps is explained in the *IMS/ESA Customization Guide*. See information about Interactive Dump Formatter in the *IMS/ESA Diagnosis Guide and Reference* for information on dialog support for offline dump formatter requests.

The FMTO options and their effect on the dumps produced are summarized in Figure 7 on page 293.

When using this figure, it is important to remember the following:

- If a SYSMDUMP DD statement is present, a machine readable dump is generated for terminating errors.
- If IMS online formatting is requested and a SYSABEND or SYSUDUMP DD statement is present, an IMS online formatted dump is generated for terminating errors.
- If IMS online formatting is not requested and a SYSABEND or SYSUDUMP DD statement is present, an MVS online formatted dump is generated for terminating errors.

Following are some additional considerations if SYSMDUMP is used:

- The SYSMDUMP DD statement should specify DISP=MOD.
- If DISP=MOD is used, then the SYSMDUMP data set must be refreshed (for example, scratched and reallocated) before it can be reused. If this is not done, new dumps are appended to the end of the data set and might not be accessible to the Offline Dump Formatting utility.

FMTO Value	IMS Online Formatting SYSABEND SYSUDUMP			MVS Online Formatting SYSABEND SYSUDUMP		IMSOOffline Formatting SYSMDUMP	
	TE	NT	FDDL	TE	NT	TE	NT

D	Y	—	Y	Y	—	Y	—
X	Y	—	Y	Y	—	Y	—
M	N	—	N	Y	—	Y	—
R	N	—	N	Y	—	Y	—
T	Y	—	Y	Y	—	Y	—
P	Y	—	N	Y	—	Y	—
F	Y	—	N	Y	—	Y	—
N	N	—	N	Y	—	Y	—
Z	N	—	N	Y	—	Y	—

TE Error would terminate IMS.

NT Error would **not** terminate IMS.

FDDL Formatted dump delete list processing.

Y Yes

N No

— Not applicable, or not attempted.

Notes:

- The default is FMTO=D.
- SYSABEND, SYSUDUMP, and SYSMDUMP are mutually exclusive.
- MVS formatting always accompanies IMS online formatting.

Figure 7. Batch FMTO Options

IRLM Use of SDUMP

The IRLM uses the MVS SDUMP program for dumping whenever its ESTAE or FRR routines are entered. SDUMP dumps are directed to the SYS1.DUMPxx data sets and printed using the IPCS service aid. SDUMP has the same advantages over the SYSABEND dump program as the spinoff dump program (explained in the previous section). The operator is told which data set contains the SDUMP by message IEA911E COMPLETE DUMP ON SYS1.DUMPxx.

Following are sample control statements for executing the IPCS program.

```
//SYSIN DD *
NEW DUMP DD=INPUT
FORMAT
LOGDATA
VTAMMAP
IRLM ir1m.subsystem.name
PRINT JOBNAME=(ir1m.job.name)
END
```

IRLM Use of SDUMP

Related Reading: For a description of the IPCS utility, see *MVS/ESA Interactive Problem Control System (IPCS) Users Guide*.

When running the IPCS program, you should provide a JOBLIB OR STEPLIB DD statement pointing to the library with the IRLM PRDUMP formatting routine (DXRRLM50). The IRLM control statement invokes this formatting routine to format IRLM control blocks.

Chapter 6. Installing the Transport Manager Subsystem

To install the transport manager subsystem, place the initial command stream into a data set referenced by the transport manager's SYSIN DD statement. This allows you to have the component start be an automated process. The JCL procedure must be tailored to your local conventions and placed in a library accessible by the MVS START command. The following is an example of the job control statements for the transport manager.

```
//TMS      PROC SYS2=,CMD=TMSTART
//          EXEC PGM=DFSMVCR0,PARM='ELX'
//STEPLIB DD DISP=SHR,DSN=IMS.&SYS2.RESLIB
//SYSIN   DD DISP=SHR,DSN=IMS.&SYS2.PROCLIB(&CMD)
//SYSOUT  DD SYSOUT=&SOUT.
//SYSPRT DD SYSOUT=&SOUT.
//SYSABEND DD SYSOUT=&SOUT.
//*
```

Figure 8. Transport Manager Job Control Statements

The transport manager subsystem requires no offline definition. All TMS definition can be performed when you start the TMS, or while the TMS is running, using transport manager commands. However, some VTAM definition is required before you can use the transport manager subsystem.

You cannot name the TMS the same name as the TM start-up member in SYS1.PROCLIB.

VTAM Definition for the Transport Manager Subsystem

In order for the transport manager subsystem to use VTAM, certain VTAM definition work is required. Because the VTAM publications describe VTAM definition specifics, the examples given here show the relationships between products and indicate which pieces of VTAM definition are important.

Related Reading: See *IMS/ESA Administration Guide: Transaction Manager* for related information.

Transport Manager

```
*****
* SYS1.VTAMLST APPL definition for Transport Manager *
* Corresponds with a Transport Manager command of: *
* SET APPLID(TMP2) APPLCOUNT(9) *
* *
*****
          VBUILD TYPE=APPL
TMP2001  APPL  EAS=64,PARSESS=YES,SRBEXIT=YES,          C
          APPC=YES,AUTOSES=1,DMINWNL=32,DMINWNR=32,DSESLIM=64,  C
          MAXPVT=1024K,VPACING=10,VTAMFRR=NO,          C
          AUTH=(VPACE),MODETAB=MODELUTM,DLOGMOD=TMDEFLT
TMP2002  APPL  EAS=64,PARSESS=YES,SRBEXIT=YES,          C
          APPC=YES,AUTOSES=1,DMINWNL=32,DMINWNR=32,DSESLIM=64,  C
          MAXPVT=1024K,VPACING=10,VTAMFRR=NO,          C
          AUTH=(VPACE),MODETAB=MODELUTM,DLOGMOD=TMDEFLT
          .
          .
          .
TMP2009  APPL  EAS=64,PARSESS=YES,SRBEXIT=YES,          C
          APPC=YES,AUTOSES=1,DMINWNL=32,DMINWNR=32,DSESLIM=64,  C
          MAXPVT=1024K,VPACING=10,VTAMFRR=NO,          C
          AUTH=(VPACE),MODETAB=MODELUTM,DLOGMOD=TMDEFLT
```

Figure 9. Sample VTAM APPL Definition

APPL

In SYS1.VTAMLST, a group of application program minor nodes must be defined for use by transport manager and IMS systems. An example is shown in Figure 9. This definition is performed with the APPL definition statement. The “name” on the APPL statement corresponds to the transport manager SET command APPLID (VTAM application id) name. Each APPL statement must have three numeric characters (beginning with 001) appended to the one to five character name specified in the SET APPLID command. The number of APPL names defined must exceed by at least one the maximum number of IMS subsystems and ILS instances expected to execute concurrently on this CPC.

Other key specifications for APPL are:

- AUTH=VPACE to allow session pacing.
- MODETAB must specify the member name of a logon mode table in SYS1.VTAMLIB that contains entries appropriate for transport manager.
- EAS specifies the number of VTAM sessions that each ACB can have. It is defined as follows:

$$2 \times (\text{maximum number of concurrent active IMS subsystems} + \text{maximum specification of the MAXCONV value}) + 2$$

A value of 64 should be more than enough.

- ENCR=NONE should be specified unless you want VTAM encryption.
- MAXPVT=1024 allows VTAM to use one megabyte of transport manager or IMS private area storage.
- PARSESS=YES allows parallel sessions and is required.
- PRTCT correlates with “vtam-acb-password” on the transport manager SET PASSWORD command.
- SRBEXIT=YES must be specified.
- VPACING=n (start with 10) to limit VTAM buffer usage while allowing reasonable bandwidth.
- VTAMFRR=NO must be specified.

MODEENT

Defines an entry in a mode table when combined with MODETAB and MODEEND macros, assembled and link edited into SYS1.VTAMLIB. An example is shown in Figure 9 on page 296. For the transport manager subsystem, the primary value is to obtain a class of service for a conversation, thus choosing a network route.

- LOGMODE specifies the logon mode table entry name, correlating to DLOGMOD on the VTAM APPL definition or to MODENAME specified by a transport manager component
- COS specifies the name of an entry in the class of service table
- SRCVPAC should be zero so that APPL VPACING= will control pacing on the PLU-SLU sessions
- SSNDPAC should be non-zero so that APPL VPACING= will control pacing on the SLU-PLU sessions
- TMDEFLT must be defined as the default logon mode table entry name

COS

Defines an entry in a class of service table when combined with COSTAB and COSEND macros, assembled and link edited into SYS1.VTAMLIB. The recommended transmission priority to specify with virtual routes is 2.

If you have enough hardware connectivity between sites, use multiple MODEENT entries and more than one class of service to select different virtual routes, thus spreading different transport manager conversations among the hardware.

COS could be used to avoid satellite links (because of latency), particularly for LOGGER conversations.

IOBUF

Defines the size and number of VTAM I/O buffers. Both the size and number have a significant impact on processor consumption and effective bandwidth. 1 KB buffers provide much better performance than small buffers, but will require more storage than small buffers where network traffic consists of many small messages. The average RU size for logger to log router conversations will be close to the OLDS blocksize (typically 22 KB).

Below is a sample VTAM mode table definition for the transport manager subsystem.

Chapter 7. Tailoring the IMS System to Your Environment

In this Chapter:

- “Tailoring the IMS Procedure Library”
- “Specifying IMS PROCLIB Members” on page 304

Tailoring the IMS Procedure Library

The results of Stage 1 sysgen definition include a set of updates to IMS.PROCLIB which are listed in the Stage 1 output. Stage 2 applies these updates to the library. You can tailor the contents of the members either before stage 2 or by direct maintenance against IMS.PROCLIB. Remember that you can set up default values for the parameters, but you can also specify them again at region startup with the symbolic parameters. Initially, these parameters use values preset by system definition, but individual region control or tuning recommendations can override the initial or default values.

Procedures Generated by System Definition

The generated members of IMS.PROCLIB fall into several categories depending on their use, as shown in Table 20.

Table 20. Categories of Generated IMS Members

Category	Member	Usage
Online Operation	IMS	Execute the control region
	IMSRDR	Invoke IMS
	DBC	Execute DBCTL region
	DBRC	Execute the DBRC address space
	DFSDCxxx	Specify data communication options
	DFSFDxxx	Execute the FDBR region
	DLISAS	Execute the DL/I address space
	DFSMPR	Execute message region
	IMSMMSG	Execute DFSMPR
	IMSBATCH	Execute batch message region
	DFSWTnnn	Print IMS spool data sets
	IMSWTnnn	Execute DFSWTnnn
	IMSFP	Execute Fast Path dependent region
	FPUTIL	Execute DEDB utilities online
	DXRJPROC	Execute IRLM region
	Online Initialization	DFSINTxx
DFSPBIMS		DB/DC execution parameters
DFSPBDBC		DBCTL execution parameters
DFSPBDCC		DCCTL execution parameters
DFSSPMxx		Storage pool management parameters
DFSVSMxx		Database pools, logging attributes
DFSDFxxx		List defining DREF requirements
DFSFIXxx		List for page fixing blocks and modules
DFSMPxxx		Module preload members
DFSSQxxx		Shared message queue and CQS address space parameters
DFSVFLxx		Modules managed by virtual fetch
INITMOD		Initialize the IMS.MODSTAT data set
DFSHSBxx		XRF complex

Tailoring IMS Library

Table 20. Categories of Generated IMS Members (continued)

Category	Member	Usage	
Online Preparation and Maintenance	ACBGEN	Maintain ACBLIB	
	SECURITY	Execute Security Maintenance utility	
	IMSDALOC	Maintain Dynamic Allocation	
	MFSUTL	Maintain MFS library	
		Other procedures are: MFSBTCH1/2, MFSTEST, MFSBACK, and MFSREST	
	MFSRVC	Maintain MFSDCT and default format	
	MFSDCT	Maintain MFS service and index	
	IMSMSV	Offline utility for MSC	
	OLCUTL	Online Change utility for library maintenance	
	Batch Operation	DBBBATCH	Stand-alone batch DL/I execution
		DLIBATCH	Stand-alone batch DI/I execution with block building
	Alternative Batch Execution	IMSPLIGO IMSCOBGO	Compile, link-edit, and execute under batch, a PL/I program (IMSPLIGO) or a COBOL program (IMSCOBGO)
Database Definition and Access	DBFMSTDBx	Specify MSDBs	
	DBDGEN	DBDLIB maintenance	
	FPUTIL	Execute online DEDB utilities	
	MFDBDUMP	Program to list the sample databases	
	MFDBLOAD	Batch execution of a program to load the sample databases	
Application Program Preparation	PSBGEN	PSBLIB maintenance	
	IMSCOBOL	Compile and link-edit COBOL program	
	IMSPLI	Compile and link-edit PL/I program	
	CBLTDLI	Link-edit control statements for COBOL	
	PLITDLI	Link-edit control statements for PL/I	

IMS automatically generates members of IMS.PROCLIB each time you perform a system definition. If you have made any changes to a prior generated library that you want to save, specify PROCLIB=NO on the IMSGEN macro.

If you perform an ONLINE, CTLBLKS, or NUCLEUS generation and have made changes affecting line groups, unit addresses, or spool output data sets, specify PROCLIB=IMS on the IMSGEN macro to replace only the member IMS.

Altering the IMS Procedure Library

Many of the members generated in IMS.PROCLIB require alteration before they can be used in direct execution of the online system, yet they provide a convenient start to the task of defining execution JCL.

You should carefully examine the members generated as a result of your system definition.

Renaming PROCLIB Members

Rename the IMS.PROCLIB members for your installation's requirements. The names can follow a convention that suggests ownership by a particular application system or a convention that has an implied sequence. For example:

IMSCTL	Control region startup (IMS renamed)
IMSMSG1	Message region startup (IMSMSG)
IMSMSG2	Second message region when required (IMSMSG)
IMSWT000	Spool output print procedure (named by IMS)

The system operator or MTO uses the procedure names to invoke the MVS job execution.

To develop these members, you need to either rename them in IMS.PROCLIB or create new ones. You can add the procedures to SYS1.PROCLIB. One option of the NODE keyword on the IMSGEN macro allows you to substitute an alternative library for the members.

Initializing Your Procedure Library

You must adjust the JCL content of the members of the procedure library. The updates you apply follow naming conventions for your IMS systems and the required DD statements. The checklist shown in Table 21 summarizes the tailoring actions, excluding arrangements for batch executions. In all procedures, you might want to add JCL comment statements for additional documentation.

Table 21. Tailoring Actions for PROCLIB Members for Batch Executions

PROCLIB Member	Tailoring Action
IMS	<ul style="list-style-type: none"> • Check and override EXEC PARM parameter. • Specify region size again. • Move to SYS1.PROCLIB. • Add database DD statements. • Specify logging data sets again; allow for dynamic allocation.
IMSRDR	<ul style="list-style-type: none"> • Adjust all symbolic parameters for message region start by system operator. • Move to SYS1.PROCLIB.
DBC	<ul style="list-style-type: none"> • Check and override EXEC PARM parameter. • Specify region size again. • Move to SYS1.PROCLIB. • Add database DD statements. • Specify logging data sets again, allow for dynamic allocation.
DBRC	<ul style="list-style-type: none"> • Add RECON DD statements, if not dynamically allocated. • Rename the procedure, if desired, and specify this name with the DBRCNM parameter of the IMS procedure. • Adjust procedures JCLOUT and JCLPDS. • Move to SYS1.PROCLIB.
DFSDCxxx	Specifies data communication options.
DFSFDRxx	<ul style="list-style-type: none"> • Defines the IMSID tracked by the active FDBR. • Defines whether FDBR applies to DEDB areas defined as SHARELVL=0 1. • Defines XCF group name for the active IMS system and the FDBR tracking region. • Specifies the number of seconds before FDBR goes into time-out status.

Tailoring IMS Library

Table 21. Tailoring Actions for PROCLIB Members for Batch Executions (continued)

PROCLIB Member	Tailoring Action
DLISAS	<ul style="list-style-type: none"> • Add full-function database DD statements and remove the corresponding statements from the IMS procedure. (No changes are necessary for dynamically allocated databases). • Specify sizes of two PSB pools using the DLIPSB and CSAPB parameters of the IMS procedure. • Move the pools to SYS1.PROCLIB. • If the procedure is renamed, specify this name with the DLINM parameter of the IMS procedure.
IMSMMSG	<ul style="list-style-type: none"> • Adjust JOB statement parameters for DFSMPR execution. • Move to IMS.JOBS
DFSMPR	<ul style="list-style-type: none"> • Check and specify EXEC PARM parameters.
IMSBATCH	<ul style="list-style-type: none"> • Check and specify EXEC PARM parameters. • Add MVS DD statements.
DFSVSMxx	<ul style="list-style-type: none"> • Define buffer pools and other options. • Allow use of sequential buffering. • Specify initial WADS and OLDS and dual logging. • Coordinate with IMS procedure.
DFSHSBxx	<ul style="list-style-type: none"> • Establish IMS active and alternate subsystems in XRF complex.
DFSDRFxx	<ul style="list-style-type: none"> • Define DREF requirements list.
DFSFIXxx	<ul style="list-style-type: none"> • Define page fix list.
DFSINTxx	<ul style="list-style-type: none"> • Define, load, and execute user-written exits during dependent region initialization.
DFSMPLxx	<ul style="list-style-type: none"> • Define preload program list.
DFSSQxxx	<ul style="list-style-type: none"> • Define PROCLIB member for CQS address space. • Define subsystem name for CQS address space. • Define name for shared expedited message handler queues. • Define primary structure containing shared message queues. • Define XCF IMS shared queues group name.
DFSVFLxx	<ul style="list-style-type: none"> • Define list of modules managed by virtual fetch.
IMSWTnnn	<p>Adjust JOB statement parameters for DFSWTnnn execution. Move to IMS.JOBS.</p>
DFSWTnnn	<ul style="list-style-type: none"> • Check and specify SYSOUT class for spooled output.
DXRJPROC	<ul style="list-style-type: none"> • Establish procedures for each required IRLM address space
DFSPBIMS	<ul style="list-style-type: none"> • Rename to DFSPBxxx, where xxx is determined by IMS procedure RGSUF=xxx. • Define IMS control region execution parameters.
DFSPBDBC	<ul style="list-style-type: none"> • Rename to DFSPBxxx, where xxx is determined by DBC procedure RGSUF=xxx. • Define DBC control region execution parameters.

Table 21. Tailoring Actions for PROCLIB Members for Batch Executions (continued)

PROCLIB Member	Tailoring Action
DFSPBDCC	<ul style="list-style-type: none"> Rename to DFSPBxxx, where xxx is determined by DCC procedure RGSUF=xxx Define DCC control region execution parameters.

Preparing for IMS Job Execution

Before executing the IMS procedure as a system task:

- Move the following procedures to SYS1.PROCLIB:
 - IMS
 - IMSRDR
 - DBC
 - DBRC
 - DLISAS
 - CQS
- Enable message region and spool output jobs:
 - Tailor IMSMSG for message region to your requirements
 - Tailor IMSWTnnn for spool output to your requirements
- Move IMSMSG and IMSWTnnn to IMS.JOBS (concatenated with SYS1.PROCLIB).
- Start the message regions and spool output jobs using one of the following:
 - IMS commands
 - System commands from the system console

Controlling Procedure Library Modifications

Using commands to name a procedure keeps the initial operating instructions simple and avoids complex symbolic parameter data entry; however, you must control the content of the JCL residing in IMS.PROCLIB in this case. For example, if the MTO enters:

```
/START REGION IMSBCH1
```

the procedure IMSBCH1 must be correctly coordinated to a known BMP, to an appropriate PSB and transaction queue and to other system options and identifying parameters.

Your control responsibilities might include auxiliary procedures for database reorganization, recovery, or system output control; furthermore, you must coordinate the modification level of each procedure to the actual production environment. For example, if an application program is modified and requires more dependent region storage, you must coordinate the program library and IMS.PROCLIB changes. A suggested technique is to include JCL comment statements that document the date and the reason for the change.

Be careful in making physical changes in procedure library members. Many of the DD statements extend over several input records and involve positional parameters. You can use a data dictionary to record and maintain the procedure library members. You can then check the changes at the terminal or review printed listings of the changed members.

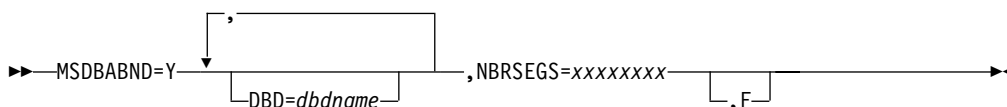
Specifying IMS PROCLIB Members

Member DBFMSDBx

This member is not applicable in the DBCTL environment.

This procedure allows you to specify the Main Storage Databases (MSDBs) to be loaded at startup. You must have previously placed this procedure in the library whose DD name is PROCLIB. Several DBFMSDBx procedures can be present in the library; differentiate them by varying the last character (the suffix, x) of the name. The MSDB= keyword parameter of the IMS procedure specifies the suffix of the particular procedure to be used at startup.

A DBFMSDBx procedure consists of one or more 80-character records, each specifying an MSDB to be loaded. The format is:



MSDBABND=

Specifies that the IMS control region is to abend if an error occurs during MSDB loading at system initialization. The valid values for the MSDBABND= parameter are:

- A** Valid for both I and C options.
- B** Valid for both Y and C options.
- C** Abend if initial checkpoint for MSDBs cannot occur.
- I** Abend for one or more of the following reasons:
 - No segments exist in the MSDBINIT data set for at least one defined MSDB.
 - For the reasons described in option Y above.
- Y** Abend if the MSDBs cannot be loaded due to errors with the MSDBINIT data set.

DBD=

Specifies the same one- to eight-character DBD name as was specified at DBDGEN. Multiple DBD= parameters are allowed.

NBRSEGS=

Specifies the number of database records expected for this MSDB. You must specify a one- to eight-digit number greater than or equal to the number of MSDB segments loaded at restart. By specifying a number greater than the number of segments to be loaded, the parameter can be used to reserve space for a terminal-related dynamic MSDB.

Literal *F* optionally specifies that the MSDB is to be page-fixed.

You can separate keyword parameters with either commas or blanks. Positions 73 through 80 are ignored but can be used to sequence the records. Syntax errors are identified by messages.

At startup, the MSDBs specified by a DBFMSDBx procedure are loaded from one of the sequential data sets identified as MSDBINIT, MSDBCP1, or MSDBCP2. You cannot, therefore, use the DBFMSDBx procedure to specify MSDBs that are not included in these data sets; an abend occurs if you use DFMSDBx in this manner.

Member DFSDCxxx

The DFSDCxxx PROCLIB member applies to the DB/DC and DCCTL environments. You can find an example of the DFSDCxxx PROCLIB member, DFSDC000, in library DFSSLIBA.

Specifying the Data Communications Options

The DC= parameter in the IMS or DCC startup procedure defines the DFSDCxxx PROCLIB member that is used. The default suffix is 000.

Control Statements

A DFSDCxxx member consists of one or more 80-character records. You code keyword parameters in positions 1-71. Positions 72-80 are ignored. Keywords can contain leading and trailing blanks. If you specify multiple keywords in a record, use commas as delimiters.

APPCASY=Y|N

Specifies whether a non-response transaction originating from a program-to-program switch should be scheduled asynchronously. The default is APPCASY=Y. IBM recommends that you keep this specification.

APPCIOT=

Specifies the APPC time-out value expressed in minutes. Valid values are between 00 and 1440. If APPCIOT=00, time-out detection is inactive.

BMPUSID=PSBNAME|USERID

Specifies the system option for the value to be placed in the user ID field of the message prefix for a message generated by a non-message driven BMP.

- If BMPUSID=USERID is specified, the value from the USER= keyword on the JOB statement is used.
- If USER= is not specified on the JOB statement, the program's PSB name is used.
- If BMPUSID=PSBNAME is specified, or if BMPUSID= is not specified at all, the program's PSB name is used.

GRAFFIN=IMS|VTAM

Specifies whether IMS or VTAM should manage generic resource affinities.

IMS indicates that IMS manages the generic resource affinities. This is the default.

VTAM indicates that VTAM manages the generic resource affinities. However, VTAM still requires IMS to manage LU6.1 (ISC) affinity.

Note that if you enter the keyword or parameter for this option incorrectly, IMS uses the default and does not generate an error message.

GRESTAE=Y|N

Specifies whether IMS should bypass the VTAM generic resource logic in the IMS ESTAE exit.

Y indicates that IMS should continue the existing ESTAE logic to delete affinity for all nodes where no status remains before closing the ACF/VTAM ACB.

DFSDCxxx

N indicates that IMS should close the ACF/VTAM ACB immediately to expedite IMS termination, and leave affinity for all nodes set.

PMTO=name

Specifies the node name of the primary master terminal (PMTO). This name overrides the PMTO node name specified during IMS system definition. If you are using a BTAM PMTO, and specify this parameter, it is ignored.

PMTO1-PMTO8=(name<,MASTER>)

Specifies the first through the eighth LTERM names of the PMTO. The names you specify here, override the first through the eighth LTERM names specified during IMS system definition. The LTERM name specified with the MASTER designation becomes the PMTO LTERM. These LTERM names must be unique for each IMS system within a shared IMS environment.

PMTOG=name

Specifies the generic LTERM name of the PMTO. The default is DFSPMTO. Use the same generic LTERM name for each IMS system within a shared IMS environment.

SMTO=name

Specifies the node name of the secondary master terminal (SMTO). This name overrides the SMTO node name specified during IMS system definition. If you are using a BTAM SMTO, and specify this parameter, it is ignored.

SMTO1-SMTO8=(name<,MASTER>)

Specifies the first through the eighth LTERM names of the SMTO. The names you specify here, override the first through the eighth LTERM names specified during IMS system definition. The LTERM name specified with the MASTER designation becomes the SMTO LTERM. These LTERM names must be unique for each IMS system within a shared IMS environment.

SMTOG=name

Specifies the generic LTERM name for the secondary master terminal. The default generic SMTO LTERM name is DFSSMTO. Use the same generic LTERM name for all IMS systems within a shared IMS environment. resource group.

TRUNC=Y|N

Specifies the system default for the truncated data options used for IMS conversational transactions. If a transaction definition (TRANSACTION) does not specify a truncated data option, the system default is used when a conversation starts for the first time.

Related Reading: For more details about truncated data, see *IMS/ESA Customization Guide*.

Restrictions:

- If you use the DFSDCxxx member to override a default generic name, you must not specify a statically-defined node name.
- The node that you specify using the PMTO or SMTO parameters must have the same physical characteristics as the default node that you are overriding. For example, you cannot override a SLU2 device with a non-SNA device.
- If you change the DFSDCxxx member, you must perform a cold start of IMS.

Member DFSDRFnn

This member is not applicable in the DBCTL environment.

Defining DREF Requirements in the IMS Virtual Control Region

You can tune your system to a particular operating environment by specifying that portions of the control region be placed in DREF storage during initialization. You must place control information in the PROCLIB procedure library (usually named IMS.PROCLIB), as part of member DFSDRFnn.

With this member, VTCBs in DREF storage are never paged out to auxiliary storage; they remain in real or expanded storage. DFSDRFnn requires that the system have sufficient expanded storage available for the VTCBs. If you choose the DREF option for VTCBs during installation, but do not have expanded storage, the VTCBs DREF storage are page fixed.

If the VTCBs have been paged out of real storage when a checkpoint occurs, DFSDRFnn can improve the time to bring them back into real storage.

The *nn* suffix of DFSDRFnn is taken from the IMS procedure FIX=*nn* initialization parameter, the same one used for the page fixing member, DFSFIXnn (see "Member DFSFIXnn" on page 309).

If you specify either DFSDRFnn or DFSFIXnn, define the one you do not choose to use as a null member. For example, if you specify FIX=55, define both members (DFSDRF55 and DFSFIX55), and make the member you do not need a null member containing a blank control statement. Without both defined, IMS issues error message DFS0579, indicating a member could not be found. If DFSFIXnn is defined as a null member, message DFS0757 is issued, reminding you that no fixing requests were made.

If BLOCK=VTCB is requested in DFSFIXnn and in DFSDRFnn, the page-fix request in DFSFIXnn is ignored.

Control Statements: The control information is contained in 80-character records. Continuation or sequencing must not be entered. The format is as follows:

►►—BLOCKS=—VTCB—◄◄

BLOCKS=

VTCB

Specifies that VTCBs are loaded into DREF storage, but are never to page to auxiliary storage.

Member DFSFDRxx

This member is applicable to the FDBR region and the FDBR-capable IMS or DBC regions.

Specifying the FDBR Options:

The DFSFDRxx member of IMS.PROCLIB specifies the FDBR options used by the FDR. Multiple DFSFDRxx members can be present in IMS.PROCLIB. The FDRMBR= parameter in the FDR, IMS, or DBC procedure identifies the member currently in effect.

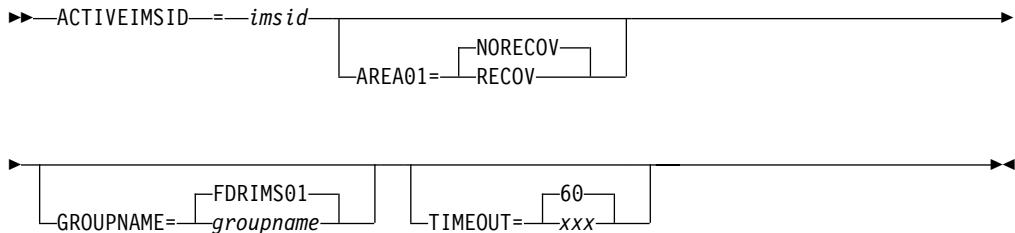
Control Statement: The record format is

Position	Contents
1-71	Option parameters separated by commas

72-80 Ignored

The first parameter in a record can have leading blanks. The last parameter in a record is denoted by a comma, followed by one or more blanks. The last parameter in the last record is followed by one or more blanks.

The following parameters can occur in any order in the control statement. Multiple control statements can be provided.



ACTIVEIMSID=

Specifies the IMS ID for the subsystem that the active FDBR region tracks. For IMS and DBC procedures (FDBR capable regions) ACTIVEIMSID must match the IMS ID defined in the procedure. This parameter is required.

AREA01=

Specifies whether Fast DB Recovery applies to DEDB areas defined as SHARELVL= 0|1. This parameter applies to FDR procedures. IMS and DBC procedures ignore it if it is specified.

RECOV

Specifies that Fast DB Recovery applies to SHARELVL= 0|1 areas.

NORECOV

Specifies that SHARELVL= 0|1 areas are not recovered by FDBR. Succeeding IMS emergency restarts recover the areas after FDBR completion.

The default is NORECOV.

GROUPNAME=

Specifies the XCF group name for the active IMS system and the tracking FDBR region. This parameter is optional. If you do not specify GROUPNAME=, a default name is created by using the ACTIVEIMSID= parameter and the prefix FDR. For example, if the active IMSID is IMS01, the default XCF group name is FDRIMS01.

TIMEOUT=

Specifies the number of seconds that FDBR waits before determining a time-out status for the tracked IMS. Valid values are from 3-999.

The default is 60.

If a value less than 3 is specified, the value 3 is used. If a value greater than 999 is specified, 999 is used.

This parameter is applicable to IMS and DBC procedures.

The value you specify with this parameter applies only to IMS time-out status from XCF. For log surveillance, FDBR uses a five-second time out value, and a three-second delay interval before calling DBRC to check whether an OLDS switch has occurred.

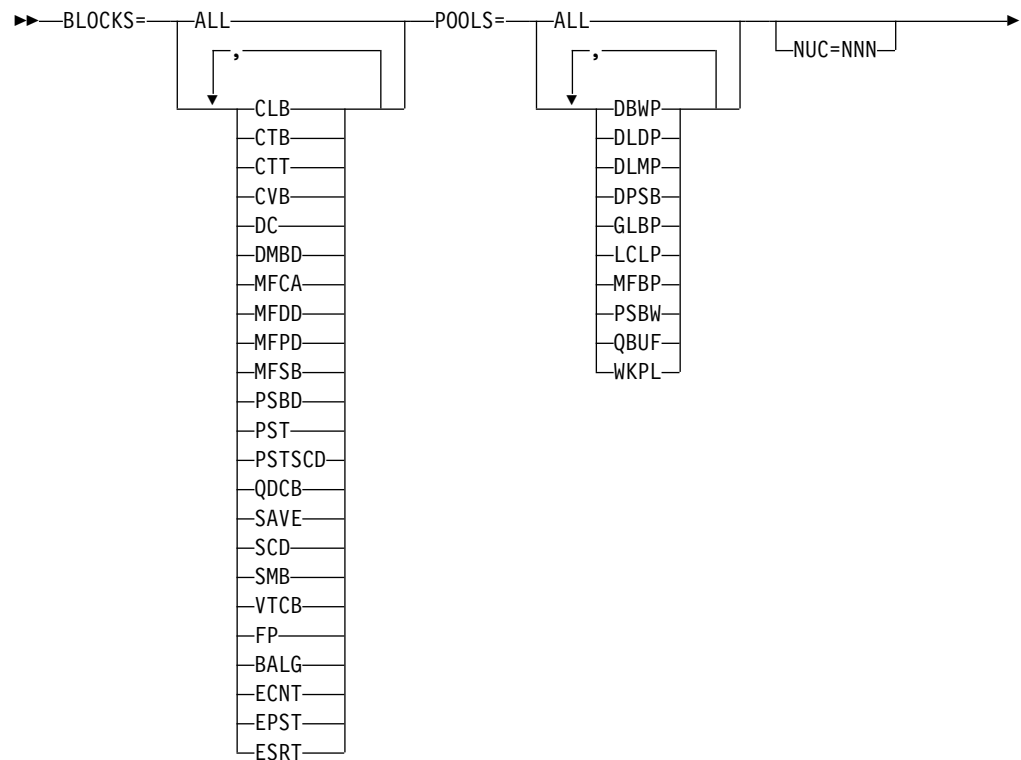
Member DFSFIXnn

Defining Fixed Pages in the IMS Control Region

You can tune your system to a particular operating environment by specifying that portions of the control region (for example, certain control blocks, buffer pools, loaded modules and part of the IMS nucleus) be fixed in address space during initialization. You must place control information in the PROCLIB procedure library (usually named IMS.PROCLIB), as part of member DFSFIXnn, where nn is a two-character field supplied in the PARM field of the EXEC statement for the control region. (See "Chapter 4. Procedures" on page 189.)

If you are running XRF, you can define an additional DFSFIXnn member. This member is identified by the DEFERFIX=xx parameter in the DFSHSBxx member of IMS.PROCLIB and can contain only non-Fast Path page-fix options for the control region. In the active system, both page-fix lists are processed. In the alternate system, the page fix list specified by the DEFERFIX=xx parameter is not processed until takeover. This allows you to control how much page fixing takes place in the alternate system while it is in tracking mode.

Control Statements: The control information is contained in 80-character records. Continuation or sequencing must not be entered. The format is as follows:





Recommendation: Specify BLOCKS=FP if you have your system has Fast Path specified by the FPCTRL macro. If you do not specify BLOCKS= either CPU performance can be seriously affected or system abends can be received, or both.

BLOCKS=

ALL

All blocks listed below

CLB¹

Non-VTAM communication line block

CTB¹

Non-VTAM communication terminal block

CTT¹

Communication translate table

CVB

Communication verb block

DC¹

Non-VTAM CLB and non-VTAM CTB

DMBD or DMBDIR

Database directories

MFCA¹

MFS buffer pool control area

MFDD¹

MFS dynamic directory, (including hash table, prime area, and up to 10 additional areas)

MFPD¹

MFS PDS directory indexes, (single area including one index per concatenation)

MFSB¹

MFS staging buffers

PSBD or PSBDIR

Program directories

PST

Dependent region PSTs

PSTSCD

Blocks PST and SCD

QDCB¹

Message queue DCBs

SAVE

Save area prefixes

SCD

System Contents Directory

SMB¹

Scheduler message blocks

VTCB¹

VTAM terminal control blocks

FP

BALG, ECNT, EPST, and ESRT

BALG¹

Load balancing group control blocks

ECNT¹

Extension to the Communication Node Tables (CNTs)

EPST

Extension to the Partition Specification Tables (PSTs)

ESRT¹

Message retrieve buffers

Note: ¹This keyword is not applicable in the DBCTL environment.

The following Fast Path control blocks, which are always page fixed, are not included in the preceding list:

BHDR

Main storage database headers

DMHR

Database buffer headers

LBUF

Synchronization point log buffer

DEDB

Control blocks for data entry databases: data management control blocks (DMCBs) and data management area control blocks (DMACs)

ESCD

Extension to the System Content Directory (SCD)

OTHR

Output thread control blocks

POOLS

This is the dynamic area acquired by IMS during initialization and used for various buffer pools.

ALL

All pools listed below

DBWP

DMB work pool

DLDP

DMB pool

DLMP

With the DL/I address space option, that portion of the PSB pool in the MVS common area.

DPSB

With the DL/I address space option, the portion of the PSB pool in DL/I local storage. Specification is ignored if the DL/I address space option is not in effect.

GLBP

Following storage pools are included: WKPL, DLMP, PSBW, DLDP, and DBWP. If the DL/I address space is used, DLDP and DBWP are not included.

LCLP¹

Following storage pools are included: QBUF and MFBP. If the DL/I address space is used, DPSB, DLDP, and DBWP are also included.

MFBP¹

Message format buffer pool

PSBW

PSB work pool

QBUF¹

Message queue buffer pool

WKPL

General working pool

Note: ¹This keyword is not applicable in the DBCTL environment.

The parameters GLBP and LCLP can represent large amounts of storage. The terms GLBP and LCLP should not be interpreted as global and local storage, because the location of these pools can vary, based on whether the local storage option or the DL/I address space option is in effect. You should avoid using GLBP and LCLP; specify the pools to be fixed by name.

NUC=NNN

Is the percentage (specified by *nnn*) of the IMS load module (DFSVNUCx) to be page fixed.

If NUC=*nnn* is specified, a percentage of the nucleus, starting at location 0, is fixed regardless of what is placed there. IMS system definition places all the DC control blocks at the low end of the nucleus.

Modulename

The name of a loaded module. The specification of a module name does not cause that module to be fixed unless it has already been loaded into storage. (Module DFSVNUC0 cannot be fixed in this manner. See NUC= keyword above.)

If the DL/I address space option is being used, an attempt is made to fix a module in both the control and DL/I address spaces. If the module is loaded in only one of these address spaces, a message indicating that the module cannot be fixed is issued, in addition to the successful fix message.

Fix Syntax Rules:

- Either commas or blanks can be used to separate parameters, except BLOCKS, POOLS, and NUC, which are followed by an equal (=) sign.
- The *nnn* subparameter of NUC= can be any one-, two-, or three-digit number from 1 to 100.

- Following the BLOCKS or POOLS parameters, either ALL or any subset of the list of blocks or pools, respectively, can be coded.
- Parameters (NUC, BLOCKS, POOLS, Modulename) can be entered in any order and can be repeated as desired. If NUC is repeated, the percentage specified on the last NUC= is used. If BLOCKS or POOLS is repeated, a block or pool is fixed if it is specified in any of the BLOCKS= or POOLS= specifications.
- The subparameters of blocks and pools can be specified in any order.
- Invalid parameters are ignored, an error message is issued, and scanning continues.
- An invalid format or an invalid word (one not beginning with an alphabetic character) causes all scanning to stop and an error message to be issued. No fixing is done.
- All 80 characters in each record are available for control information. Continuation or sequencing cannot be entered.
- The modulename must be specified as the full eight-character name of a loaded module.

Example:

```
DFSDLR00 NUC=76 BLOCKS=DC,SMB,PSTSCD POOLS=MFBP
```

This example causes:

- Module DFSDLR00 to be fixed
- 76% of the IMS nucleus to be fixed
- The control program nucleus blocks DC, SMB, PST, and SCD to be fixed
- The message format buffer pool to be fixed

The member specified in the EXEC PARM field is read and processed during the IMS initialization procedure.

A list of the fixed pages is maintained during the execution of IMS. When IMS terminates, this list is used to free the fixed pages prior to returning to MVS.

Member DFSHSBxx

This member is not applicable in the DBCTL environment.

Specifying the XRF Options

The DFSHSBxx member of IMS.PROCLIB specifies the XRF options used by the active and alternate subsystems in an XRF complex. Multiple DFSHSBxx members can be present in IMS.PROCLIB. The HSBMBR= parameter in the IMS procedure identifies the member currently in effect.

An XRF complex has two DFSHSBxx members in effect at any time, one for the active system, and one for the alternate subsystem. The two members need not be identical, but the USERVAR and RSENAME parameters must be the same, and the surveillance options should be the same. The parameters can have different meanings, depending on whether the subsystem is the active or the alternate subsystem (see SURV=, on page 316).

Control Statement: The record format is:

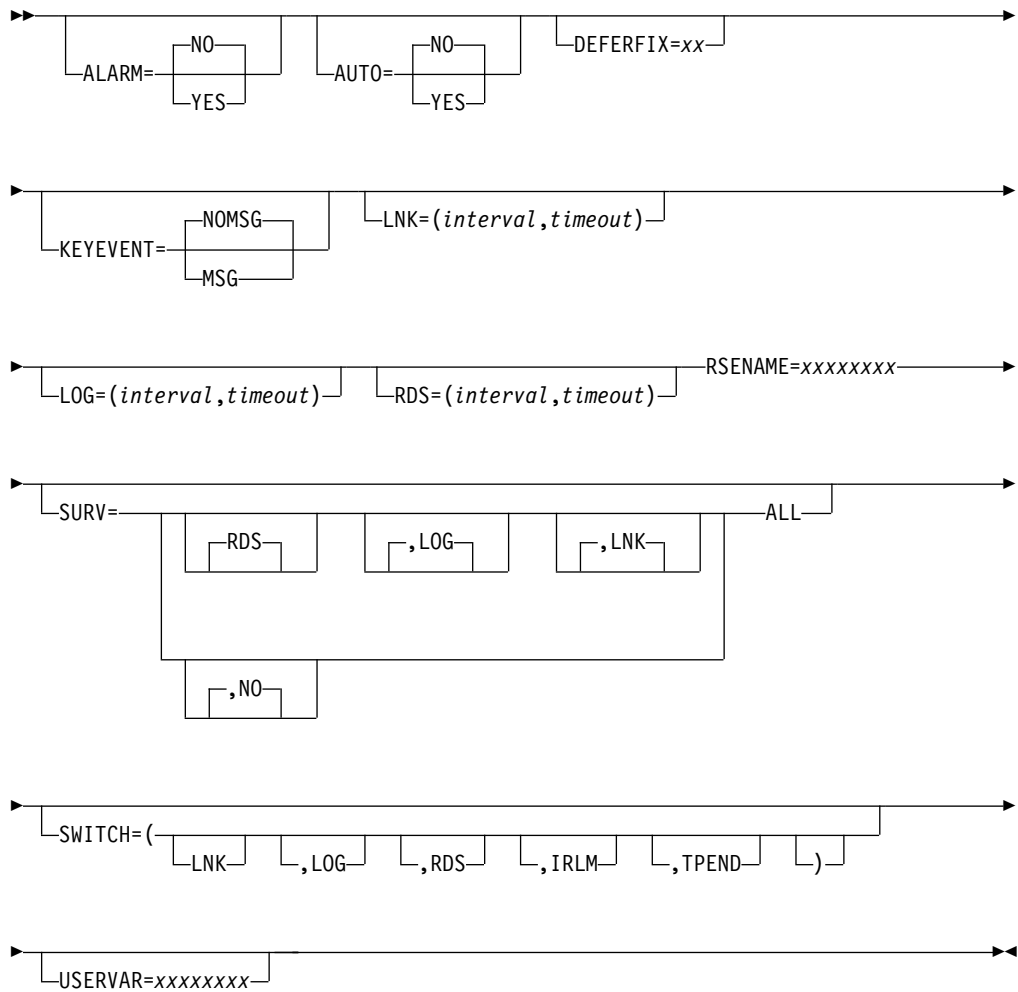
Position	Contents
1-71	Option parameters separated by commas
72-80	Ignored

DFSHSBxx

The first parameter in a record can have leading blanks. The last parameter in a record is denoted by a comma, followed by one or more blanks. The last parameter in the last record is followed by one or more blanks.

XRF Option Parameters: The following XRF option parameters can occur in any order in the control statement. Multiple control statements can be provided.

RSENAME= is required for all XRF complexes, and USERVAR= is required if you use VTAM.



ALARM=

Specifies whether (YES) or not (NO) the service processor alarm on the alternate subsystem sounds at takeover request. The default is NO.

AUTO=

Specifies whether (YES) or not (NO) a takeover is to proceed automatically. The default is NO.

DEFERFIX=xx

Identifies the DFSFIXxx member of IMS.PROCLIB containing the non-Fast Path page fixing options for the control region, which are processed during the restart of an active subsystem, and when the alternate subsystem becomes the active during takeover.

KEYEVENT=

Controls whether your IMS operator receives all XRF- related messages.

- MSG indicates that noncritical-event messages and critical-event messages are sent.
- NOMSG indicates that only critical-event messages are sent.

The default is NOMSG.

LNK=(interval,timeout)

Specifies the timing values for the ISC link surveillance. The default is (3,9).

interval

In the active subsystem, specifies how often (1 to 99 seconds) IMS sends signals across the ISC link.

In the alternate subsystem, specifies how often (1 to 99 seconds) IMS checks for those signals.

See Table 22 on page 317 for additional information on coding the interval value.

timeout

In the alternate subsystem, specifies how long (1 to 999 seconds) IMS waits for a signal before considering a takeover.

See Table 23 on page 317 for additional information on coding the timeout value.

LOG=(interval,timeout)

Specifies the timing values for the log surveillance. It also determines the frequency with which the alternate subsystem accesses the log, after it has caught up. The default is (1,3).

interval

In the alternate subsystem, specifies how often (1 to 99 seconds) IMS checks to see if a new log record is received from the active subsystem.

This parameter also determines the frequency with which the alternate subsystem attempts to read a new log record, after the alternate has caught up with the active. If you do not specify LOG surveillance or do not specify the interval value, the alternate subsystem waits one second between reads to the log (only if the alternate has caught up with the active).

See Table 22 on page 317 for additional information on coding the interval value.

timeout

In the alternate subsystem, specifies how long (1 to 999 seconds) IMS waits for a new record, before considering a takeover.

When the alternate subsystem catches up with the active, the timeout and interval values are also used to determine a delay interval, which limits the frequency of calls to DBRC to check if an OLDS switch has occurred.

See Table 23 on page 317 for additional information on coding the timeout value.

RDS=(interval,timeout)

Specifies the timing values for the RDS surveillance. The default is (1,3).

interval

In the active subsystem, specifies how often (1 to 99 seconds) IMS writes a timestamp on the RDS.

In the alternate subsystem, specifies how often (1 to 99 seconds) IMS checks for timestamps on the RDS.

See Table 22 on page 317 for additional information on coding the interval value.

timeout

In the alternate subsystem, specifies how long (1 to 999 seconds) IMS waits for a timestamp before considering a takeover.

See Table 23 on page 317 for additional information on coding the timeout value.

RSENAME=xxxxxxx

Specifies the name of the recoverable service element (RSE) containing the active and alternate IMS subsystems in an XRF complex. The RSENAME= specifications must be the same in DFSHSBxx members for the active and alternate subsystems.

The name used for the RSENAME parameter should not be the same name used for the IMSID parameter in the IMSCTRL macro for either the active or the alternate IMS subsystems in the XRF complex.

RSENAME= is required.

SURV=

Specifies the surveillance mechanisms to run in an XRF complex. The SURV= parameters in the DFSHSBxx members for the active and alternate subsystems must be the same. The default is RDS,LOG,LNK.

RDS

The alternate subsystem is to periodically check for timestamps in the RDS.

LOG

The alternate subsystem is to periodically check for new log records in the IMS log.

LNK

The alternate subsystem is to periodically check for signals coming across the ISC link.

ALL

All three surveillance mechanisms are to be used.

SWITCH=

Controls if the failure of a surveillance mechanism, VTAM, or the IRLM should cause the alternate subsystem to consider a takeover.

LNK

The absence of signals across the ISL link after a timeout period elapses causes IMS to consider a takeover.

LOG

The absence of a new log record in the IMS log after a timeout period elapses causes IMS to consider a takeover.

RDS

The absence of a new timestamp in the RDS after a timeout period elapses causes IMS to consider a takeover.

IRLM

An IRLM failure causes IMS to request a takeover.

TPEND

A VTAM failure that invokes the IMS TPEND exit routine causes IMS to request a takeover.

The default is SWITCH=(LNK,LOG,RDS),(IRLM),(TPEND), signifying IMS requests a takeover for any one of the following events.

- The LNK, LOG, and RDS surveillance mechanisms all indicate a failure.
- The IRLM fails.
- VTAM fails and causes the IMS TPEND exit routine to be invoked.

USERVAR=xxxxxxxx

Specifies the USERVAR that VTAM uses to associate a logon message with the VTAM application name (APPLID) of the current active IMS subsystem in an XRF complex. The USERVAR= in DFSHSBxx members for the active and alternate subsystem must be the same.

USERVAR= is required if you have VTAM.

Table 22. Initializing the INTERVAL Value

Subsystem	Requirement	Exception Action
Alternate	Alternate interval value must be greater than or equal to active interval value.	DFS3812 issued. Alternate interval value forced to active interval value.
Alternate	Alternate interval value must be greater than or equal to alternate LOG interval value (LNK and RDS).	DFS3833 issued. Alternate interval value forced to alternate log interval value.
Alternate	Alternate interval value less than or equal to alternate RDS and LNK interval value (LOG only).	DFS3833 issued. Alternate interval value forced to alternate RDS then LNK interval value.
Alternate	Twice the interval value must be less than or equal to the alternate timeout value.	DFS3832 issued. Alternate timeout value forced to twice the interval value.
Active	Twice the interval value must be less than or equal to the active timeout value.	DFS3832 issued. Active timeout value forced to twice the interval value.

Table 23. Initializing the TIMEOUT Value

Subsystem	Requirement	Exception Action
Alternate	Timeout value must be greater than or equal to twice the alternate interval value.	DFS3832 issued. Alternate timeout value forced to twice the alternate interval value.
Active	Timeout value must be greater than or equal to twice the active interval value.	DFS3832 issued. Active timeout value forced to twice the active interval value.

Member DFSINTxx

Specifying Dependent Region Preinitialization Routines

The DFSINTxx member of IMS.PROCLIB is used by MPR, IFP, and BMP dependent regions (PREINIT= exec parameter) to identify the preinitialization modules to receive control prior to region initialization.

Related Reading: For information on coding these routines, see *IMS/ESA Customization Guide*.

Control Statement: The record format is:

Position	Contents
1-71	Module names, separated by a comma
72-80	Ignored

A separate control statement for each module is required.

Member DFSMPLxx

Making High-Use Program Modules Resident

In addition to those modules automatically preloaded into the IMS control region, other MVS, IMS, or user-written program modules can be made resident in IMS regions by using the IMS Module Preload function. Doing so can improve throughput and response time for transactions frequently referred to if sufficient virtual storage is available with high-performance paging DASD. Ordinarily, the Module Preload function is advantageous only for MPPs and for message-driven Fast Path regions, and only when the preloaded modules are reentrant.

Program modules can be made resident in either an IMS region or LINKPACK. This section explains how to make program modules permanently resident and provide some rules for doing so.

Resident in a Region: When making modules resident in a region, consider the following:

- Serially reusable MVS, IMS, or user-written program modules only can be resident in a region.
- Only program modules used for transactions serviced by the region involved, should be made resident in that region. Program modules should be resident for only the duration of the region.
- MVS and IMS modules that execute in a dependent region can reside in that dependent region.

Modules made resident in a region are in the region JOBPack and are called without repeating the overhead of searching STEPLIB/JOBLIB, LINKPACK, and SYS1.LINKLIB. The overhead of fetching the module into virtual storage is encountered only at region initialization time.

Resident in LINKPACK: In an operating environment where there are several batch regions or a combination of online and batch regions, it can be advantageous to place some of the frequently used IMS and access method modules in the operating system LPA. Programs made resident in LPA should be those that can be shared among all regions. This saves virtual storage space. The modules to be included must exist in either a LNKLSTxx member of SYS1.PARMLIB (or its concatenations, one of which should be IMS.RESLIB) or SYS1.SVCLIB.

Initial access of LPA resident program modules can be slow, because the region JOBPACK and STEPLIB/JOBLIB are searched before LPA is searched. Subsequent access can be at CPU speeds if the region JOBPACK has not been purged by MVS virtual storage management (as when sufficient virtual storage is not available to satisfy a user GETMAIN for space). Although the modules are physically residing in LPA (and are shared among multiple regions), the overhead involved in searching program libraries and LPA is only experienced at region initialization time.

Although some real storage can be saved by putting the DL/I action modules into LPA, it is not recommended that an installation place IMS modules into the LPA. Doing so makes it difficult to execute different versions of IMS concurrently. Also, when maintenance is applied to modules in LPA, another initial program load of the MVS system is required.

The IMS Module Preload Function: The MVS task under which modules are preloaded varies based on the IMS region type.

IMS Region Type	MVS Task
Control (CTL)	Physical Log
Message (MSG)	Program Control
Batch Message (BMP)	Region/Program Control
Batch (DLI)	Region/Program Control
Fast Path (IFP)	Region/Program Control

Control Statement: Complete the following steps to use this function to make program modules resident in a region or LINKPACK.

1. Use the IMS.PROCLIB data set allocated prior to Stage 2 of IMS system definition and the MVS utility IEBUPDTE to create IMS.PROCLIB members that identify modules to be preloaded.

Related Reading: See *MVS/ESA Data Administration: Utilities*. Member names must be DFSMPLxx. The record format is:

Position	Contents
1-71	Module names, separated by a comma
72-80	Ignored

The first name on a record can have leading blanks. The last name on a record is denoted by a comma followed by one or more blanks; the last name on the last record is followed by one or more blanks.

No limit exists for the number of module names that can be specified. If a preloaded module has ALIAS names that are ordinarily invoked via MVS LINK (that is, application programs in DLI BMPs and MPPs), those ALIAS names should also be specified for preload.

2. Link-edit the application program modules using the IMS reentrant DL/I language interface. The true attributes must be specified; for example, if reusable only, do not specify "RENT."
The IMS DL/I language interface is not reentrant. Any IMS application programs that were designed to be reentrant or serially reusable can use the module preload function after being link-edited with the IMS language interface.
3. In the step execution JCL, do both of the following:
 - Insert a DD statement for IMS.PROCLIB with the ddname PROCLIB.

DFSMPLxx

- Specify the correct PROCLIB member in the EXEC statement parameter.

If an invalid member name is specified, or if the record format is incorrect, no modules are loaded but a message is issued and initialization continues.

The member name should be unique unless overridden by the IMS procedure.

For a description of the EXEC statement parameters, see “DFSMPR Procedure” on page 211.

Member DFSPBDBC

This member applies only to a DBCTL environment. It allows you to specify DBCTL control region execution parameters. The parameters that you specify in this member override those specified in the stage 1 macros.

You can place several DFSPBDBC members in PROCLIB by replacing the member name DFSPBDBC with DFSPBxxx, where xxx must be three alphanumeric characters. The RGSUF= keyword in the DBC procedure specifies the xxx suffix to be used during startup of the DBCTL control region.

The DFSPBxxx procedure consists of one or more 80-character records. Each record contains one of the following:

- A comment, marked by an '*' in statement column 1.
- One or more execution parameters in the following form:

```
KEYWORD=value
```

Parameters are separated with commas (,). Each parameter can be preceded by a space or the comma following the prior parameter value. You can have blanks on either, both, or neither side of the comma that separates keyword strings.

An example of the DFSPBDBC member is shown below. For a list of parameters and parameter descriptions, see “DBC Procedure” on page 195.

```
ARC=,  
AUTO=,
```

Member DFSPBDCC

This member applies only to a DCCTL environment. It allows you to specify DCCTL control region execution parameters. The parameters that you specify in this member override those specified in the stage 1 macros.

You can place several DFSPBDCC members in PROCLIB by replacing the member name DFSPBDCC with DFSPBxxx, where xxx must be three alphanumeric characters. The RGSUF= keyword of the DCC procedure provides the xxx suffix to be used during startup of the DCCTL control region.

The DFSPBxxx member consists of one or more 80-character records. Each record contains either:

A comment, marked by an '*' in statement column 1.

One or more execution parameters in the following form:

```
KEYWORD=value
```

The parameters are separated by commas (.). Each parameter can be preceded by a space or the comma following the prior parameter value. You can have blanks on either, both, or neither side of the comma that separates keyword strings.

An example of the DFSPBDCC member is shown below. For a list of parameters and parameter descriptions, see “DCC Procedure” on page 201.

```
ARC=,
AUTO=,
```

Member DFSPBIMS

This member applies only to a DB/DC environment. It allows you to specify DB/DC control region execution parameters. The parameters that you specify in this member override those specified in the stage 1 macros.

You can place several DFSPBIMS members in PROCLIB by replacing the member name DFSPBIMS with DFSPBxxx, where xxx must be three alphanumeric characters. The RGSUF= keyword of the IMS procedure provides the xxx suffix to be used during startup of the DB/DC control region.

The DFSPBxxx member consists of one or more 80-character records. Each record contains one of the following:

- A comment, marked by an '*' in statement column 1.
- One or more execution parameters in the following form:

```
KEYWORD=value
```

Parameters are separated by commas (.). Each parameter can be preceded by a space or the comma following the prior parameter value. You can have blanks on either, both, or neither side of the comma that separates keyword strings.

An example of the DFSPBIMS member is shown below. For a list of parameters and parameter descriptions, see “IMS Procedure” on page 221.

```
ARC=,
AUTO=,
```

Control Statements:

APPCIOT=xx

Specifies the APPC time-out value. xx is a value between 00 and 99. xx is the number of minutes an APPC call can wait without being considered timed-out. If you specify APPCIOT=00, time-out detection is inactive.

APPCAST=xx

Specifies the APPC time-out value for asynchronous conversations. xx is a value between 00 and 99. xx is the number of minutes an APPC call can wait without being considered timed-out. If you specify APPCAST=00, time-out detection is inactive.

Member DFSRSRxx

Specifying the RSR Options

The DFSRSRxx member of IMS.PROCLIB specifies the RSR options used by the online active and tracking subsystems in an RSR complex. Multiple DFSRSRxx

DFSRSRxx

members can be present in IMS.PROCLIB. The RSRMBR= parameter in the IMS, DBC, and DCC procedures identifies the member currently in effect. IMS batch jobs do not use the DFSRSRxx member.

An RSR complex has at least two DFSRSRxx members in effect at any time, one for the online active system and one for the tracking subsystem. The two members need not be identical, but the GSGNAME parameters must be the same. The parameters can have different meanings, depending on whether the subsystem is the active or the tracker.

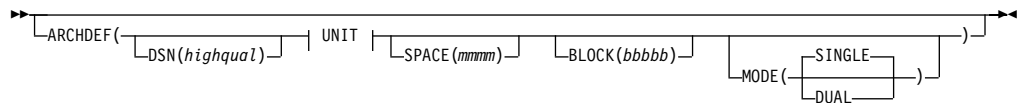
If RSRFEAT=RLT or RSRFEAT=DLT is not specified in the IMSCTRL macro during IMS system definition, the specification of a GSGNAME and TMINAME in the DFSRSRxx member is not accepted.

Control Statement: Because the parameters in the DFSRSRxx member are parsed by the TSO parser, the parameters must follow TSO parameter syntax rules. Any parsing errors are recorded in the SYSTSPRT data set. The record format is:

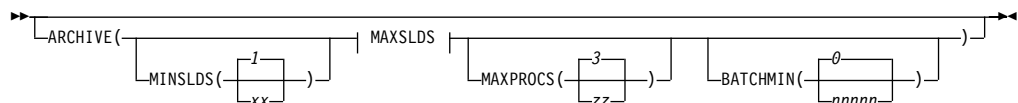
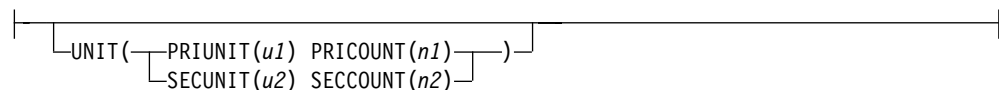
Position	Contents
1-71	Option parameters separated by commas
72-80	Ignored

The first parameter in a record can have leading blanks. Delimiters between parameters can be spaces or commas. The last parameter in a record is denoted by a comma, followed by one or more blanks. The last parameter in the last record is followed by one or more blanks. Comments are delimited by /* and */.

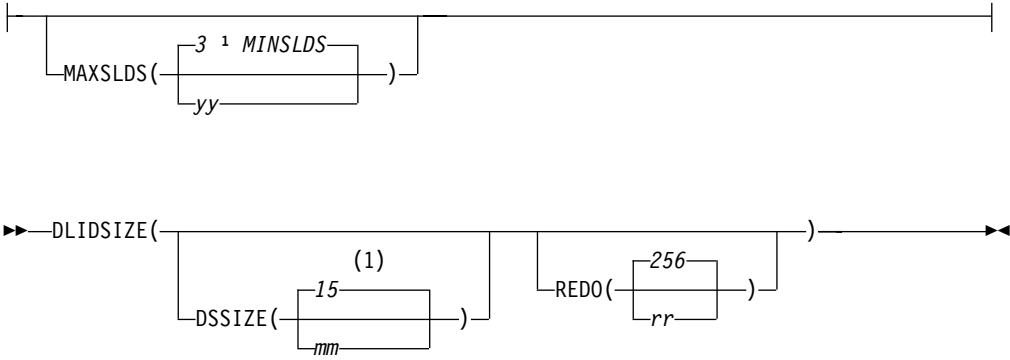
RSR Option Parameters: All of the RSR option parameters can occur in any order in the control statement. Multiple control statements can be provided.



UNIT:

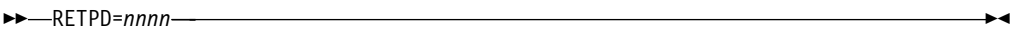
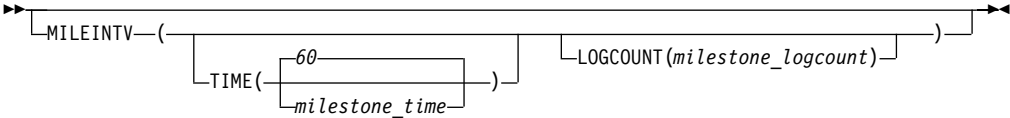
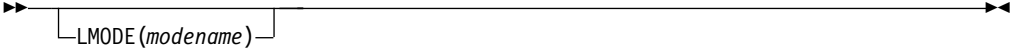
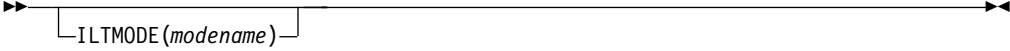
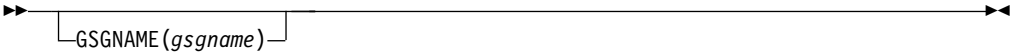
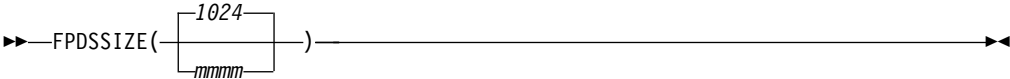


MAXSLDS:

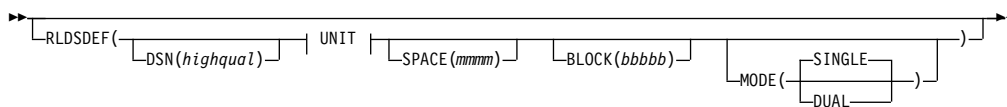


Notes:

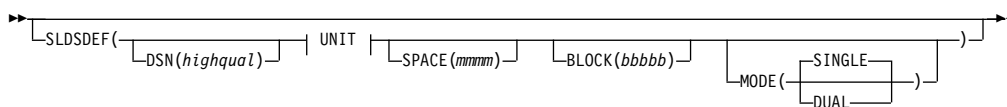
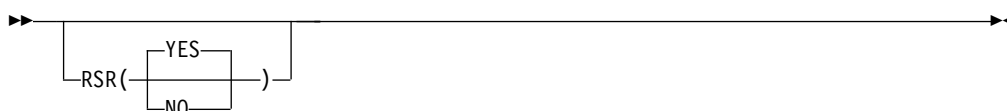
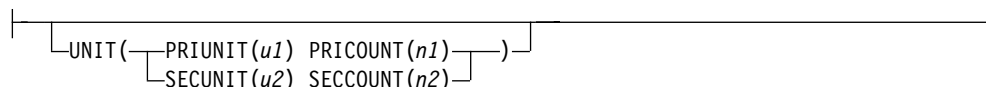
1 Or the number of PSTs, whichever is greater.



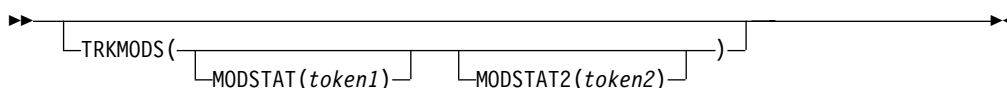
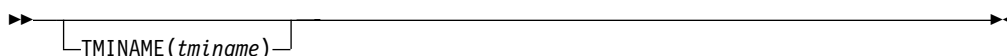
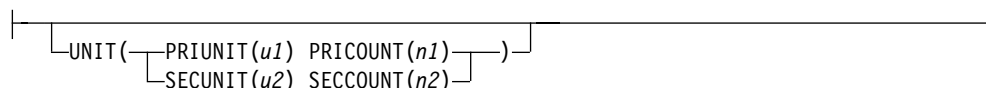
DFSRSRxx



UNIT:



UNIT:



ARCHDEF

Defines the archived secondary system log data set (SLDS) characteristics for the tracking subsystem.

If ARCHDEF is not specified, no automatic archiving occurs.

DSN

Specifies the initial qualifiers of the data set name to be used for archive data sets.

No default data set name exists.

The character string specified must satisfy data set name syntax rules and be less than or equal to 27 characters in length. The actual data set name generated looks like: highqua1.ARCHc.Nnnnnnnn, where c is 1 (one) for the primary SLDS archive and 2 (two) for the secondary SLDS archive, and nnnnnnn is a number generated to identify uniquely the data set name.

UNIT

Specifies the primary and secondary unit types to be used when creating or reading SLDS archive data sets.

PRIUNIT

Specifies the unit type to be used for the data set (if MODE(SINGLE) is specified) or for the primary data set (if MODE(DUAL) is specified). Example unit types are: SYSDA, TAPE, 3380, 3390.

No default unit type exists. You cannot specify different device types for the two units used in dual logging of a tracker log data set type.

PRICOUNT

Specifies the number of devices to be allocated for the data set or primary data set. This is most meaningful for tape devices, because multiple tapes can be mounted, thus eliminating the need to wait for tape rewind and new tape mount.

The default for PRICOUNT is zero. Zero specifies a data set consisting of up to five volumes for tape data sets. Specifying any other value for PRICOUNT specifies multi-volume data sets in a maximum of 255 volumes per data set.

The recommendation is to specify PRICOUNT(0) and SECCOUNT(0) to limit the number of volumes to five or less when allocating the archived SLDS during restart. Attention needs to be taken to ensure the amount of data being archived will fit into five volumes if tape data sets are being used.

SECUNIT

Specifies the unit type to be used for the secondary data set. Example unit types are: SYSDA, TAPE, 3380, 3390.

No default unit type exists. You cannot specify different device types for the two units used in dual logging of a tracker log data set type.

SECCOUNT

Specifies the number of devices to be allocated for the secondary data set. This is most meaningful for tape devices because multiple tapes can be mounted, thus eliminating the need to wait for tape rewind and new tape mount.

The default for SECCOUNT is 0 (zero). A specification of 0 results in data sets consisting of up to five volumes for tape data sets. Any other value for SECCOUNT results in a maximum of 255 volumes per data set.

SPACE

Specifies the space allocation in megabytes.

No default space allocation exists.

For DASD, the value specified is converted to blocks to give an integral number of cylinders. For tape units, the value specified controls switching from one data set to the next.

For tape units, if your space allocation is small enough to cause tape end-of-volume before the physical end of the tape, the data set overflows to another volume. The space allocation must be greater than the product of the values specified in SLDSDEF SPACE and ARCHIVE MINSLDS.

Formula

$$((\text{SLDSDEF SPACE}) * \text{MINSLDS} \leq (\text{ARCHDEF SPACE}))$$

BLOCK

Specifies the block size to be used for all SLDS archive data sets. No default block size exists.

This number should be a fairly large number to ensure good RSR performance. A block size of 32 KB is recommended for tape and half-track for DASD (22 KB for 3380 and 26 KB for 3390 devices). The block size must be at least as large as the largest incoming IMS log record. The range of valid values is 4096 to 32760.

MODE

Specifies whether SINGLE or DUAL archive SLDS logging is required. The default is SINGLE.

ARCHIVE

Controls automatic archiving of SLDS data sets created by a tracking subsystem. If ARCHIVE is not specified, no automatic archiving occurs.

Related Reading: See *IMS/ESA Operations Guide* for more information on how to use ARCHIVE to control automatic archiving of tracking SLDSs.

MINSLDS

Specifies the minimum number of tracking SLDSs to be archived for each active subsystem.

The range of valid values is 1 to 999. The default is 1.

MAXSLDS

Specifies the maximum number of tracking SLDSs to be maintained for each active subsystem before an archive operation will be triggered.

The range of valid values is the MINSLDS value to 999. The default is 3 times MINSLDS.

MAXPROCS

Controls the number of concurrent archive operations.

The range of valid values is 1 to 99. The default is 3.

BATCHMIN

Specifies the minimum number of log records required before the system initiates automatic archiving. BATCHMIN is valid for log records that originate from active site batch IMS jobs. If a batch IMS job generates fewer log record than specified by the BATCHMIN parameter, the tracking SLDS is not archived.

The range of valid values is 1 to 999999.

If BATCHMIN is not specified, all tracking SLDS from batch IMS jobs are automatically archived when automatic archive is active.

DLIDSIZE

Specifies how the DL/I database tracker uses its data space. The specification indicates the amount of virtual storage for the DL/I tracking data space that can be backed by real and expanded storage for those times when the database tracker cannot keep up with the active subsystems.

If you specify a number larger than the available storage for the database tracker, the machine can experience excessive paging, thus reducing the efficiency of the tracker. If you specify a number smaller than the available storage for the database tracker, the tracker might need to perform catch-up processing more often than is truly necessary.

DSSIZE

Specifies the maximum number of megabytes for DL/I database tracking data space usage.

The range of valid values is 15 to 1600. The default is either 15 or the number of PSTs (set by the PST parameter on the EXEC PARM= statement), whichever is greater.

REDO

Specifies an estimated average size, in bytes, for the active subsystem's DL/I redo (type X'50') log records. The number specified is rounded up to the nearest power of two.

The range of valid values is 128 to 4096. The default is 256.

FPDSSIZE

Specifies the threshold value size of storage, in megabytes, used for Fast Path database tracking data spaces. When data space usage reaches the value specified, the stored update data is written to disk. Actual storage size can be larger than the specified value, because additional data is normally processed while data is written to disk.

The range of valid values is 1 to 1024. The default is 1024.

GSGNAME

Specifies the one- to eight-character global service group name to be used for the particular RSR complex. This GSG name overrides any specified during system definition (in the IMSCTRL macro).

Both the active and tracking subsystems must specify the same GSG name, either during system definition or in the DFRSRxx member.

No default GSG name exists.

ILTMODE

Specifies the one- to eight-character mode name to be used by the transport manager subsystem for isolated log sender conversations.

If ILTMODE is not specified, the default logon mode name defined in the DLOGMOD of the transport manager subsystem's VTAM APPL definition is used.

LMODE

Specifies the one- to eight-character mode name to be used by the transport manager subsystem for the logger and log router conversations.

If LMODE is not specified, the TMS-implementation defined default logon mode name, TMDEFLT, is used.

MILEINTV

Specifies the interval between milestones for tracking subsystems. Milestones are used by the tracking subsystem for tracker restart. Tracker restart is not a remote takeover. They keep track of the current positions on the SLDSs containing active IMS log records and of which log records have been applied to the shadow databases.

The milestone can be triggered either by time interval or by number of log records received from all active subsystems. You can specify either or both of these methods for triggering milestones, but the timer (default or specified value) is used in all cases. If both are specified, the first one reached (timer or log record count) triggers the milestone and both counters are reset to zero.

TIME

Specifies the maximum elapsed time between milestones. The range of valid values is 10 to 600 seconds. The default is 60 seconds.

LOGCOUNT

Specifies the maximum amount of log data that can be received from all active subsystems between milestones.

The value can be specified as either *nnnK* for kilorecords of data or as *nnnM* for megarecords of data. The range of valid values is 1K to 2047M (or 2 096 128K) records. No default milestone log count exists.

MTOID

Specifies which terminals to use for the IMS master terminal and secondary master terminal. The terminals specified must be listed on the NAME= parameter of the TERMINAL macro. MTOID also controls which password is used, The password must be included on the list in the PASSWD= parameter of the COMM macro.

This parameter is only valid for tracking subsystems.

Valid values are 1, 2 or 3, depending on the number of names specified in the TERMINAL and COMM macros. The default is 1. The VTAM APPLID is overridden by the APPLIDn= startup parameter and APPLID3 is always used for the tracking subsystem.

You must take care not to cause the same IMS VTAM APPLID to be opened for more than one IMS subsystem. If necessary, the APPLID1, APPLID2 and APPLID3 parameters should all be overridden to ensure unique APPLIDs.

RETPD

Specifies the retention period, *nnnn*, for a tracking log data set, where *nnnn* is a one- to four-digit value representing a number of days. The system adds the number you specify to the current date to arrive at an expiration date. If you do not specify RETPD, the default is either the system default for this class of data set or 0. Use of this optional parameter helps reduce the possibility of accidental deletion of the tracking log data set. After the retention period

expires, another data set can write over the tracking log data set. You cannot use the RETPD parameter to change the expiration date of existing tracking log data sets if they are managed by SMS.

You can also specify the retention period of the tracking log data set using the EXPDT parameter on standard JCL. Code the RETPD parameter when you want to specify a retention period for the tracking log data set or when you want to override the retention period defined in the data class for new tracking log data sets.

The RETPD parameter cannot have a null value.

RLDSDEF

Defines the recovery log data set (RLDS) characteristics for the tracking subsystem.

No default RLDS definition exists. If RLDSDEF is not specified, no RLDS is created at the tracking site.

DSN

Specifies the initial qualifiers of the data set name to be used for RLDS data sets.

No default data set name exists.

The character string specified must satisfy data set name syntax and be less than or equal to 27 characters in length. The actual data set name generated looks like: highqual.RLDS.c.Nnnnnnn, where c is 1 for the primary RLDS data set and 2 for the secondary RLDS, and nnnnnnn is a number generated to uniquely identify the data set name.

UNIT

Specifies the primary and secondary unit types to be used when creating or reading RLDS data sets.

PRIUNIT

Specifies the unit type to be used for the data set (if MODE(SINGLE) is specified) or for the primary data set (if MODE(DUAL) is specified). Example unit types are: SYSDA, TAPE, 3380, 3390.

No default unit type exists. You cannot specify different device types for the two units used in dual logging of a tracker log data set type.

PRICOUNT

Specifies the number of devices to be allocated for the data set or primary data set. Specifying PRICOUNT is most useful for tape devices, so that multiple tapes can be mounted, thus eliminating the need to wait for tape rewind and new tape mount.

The default for PRICOUNT is 0 (zero). A specification of 0 results in data sets consisting of up to five volumes for tape data sets. Any other value for SECCOUNT results in a maximum of 255 volumes per data set.

Recommendation: Specify PRICOUNT(0) and SECCOUNT(0) to limit the number of volumes to five or less when allocating RLDS data sets. Attention needs to be taken to ensure that the amount of data being archived will fit into five volumes if tape data sets are being used.

SECUNIT

Specifies the unit type to be used for the secondary data set. Example unit types are: SYSDA, TAPE, 3380, 3390.

There is no default unit type. You cannot specify different device types for the two units used in dual logging of a tracker log data set type.

SECCOUNT

Specifies the number of devices to be allocated for the secondary data set. This is most meaningful for tape devices because multiple tapes can be mounted, thus eliminating the need to wait for tape rewind and new tape mount.

The default for SECCOUNT is zero. A specification of zero results in data sets consisting of up to five volumes for tape data sets. Any other value of SECCOUNT results in a maximum of 255 volumes per data set.

SPACE

Specifies the space allocation, in megabytes, for each RLDS data set.

The range of valid values is 1 to 4095. No default space allocation exists.

BLOCK

Specifies the block size to be used for all RLDS data sets. No default block size exists.

This value should be a fairly large number to ensure good RSR performance. A block size of 32 KB is recommended for tape archiving and half-track for DASD (22KB for 3380 and 26KB for 3390 devices). The block size must be at least as large as the largest incoming IMS log record. The range of valid values is 4096 to 32767.

MODE

Specifies whether SINGLE or DUAL RLDS logging is required. The default is SINGLE.

RSR

Specifies whether the active IMS subsystem is to be enabled for RSR processing. The default is YES.

With RSR(YES), you must also specify a GSG name and a TMI name, either in the DFRSRxx member or in the IMSCTRL macro.

SLDSDEF

Defines the shadow system log data set (SLDS) characteristics. This parameter is required for all tracking subsystems. No default for SLDS definition exists.

DSN

Specifies the initial qualifiers of the data set name to be used for SLDS data sets. No default data set name exists.

The character string specified must satisfy data set naming conventions and be less than or equal to 27 characters in length. The actual data set name generated looks like: highqua1.SLDS.c.Nnnnnnn, where c is 1 for the primary SLDS data set and 2 for the secondary SLDS, and nnnnnnn is a number generated to uniquely identify the data set name.

UNIT

Specifies the primary and secondary unit types to be used when creating or reading SLDS data sets.

PRIUNIT

Specifies the unit type to be used for the data set (if `MODE(SINGLE)` is specified) or for the primary data set (if `MODE(DUAL)` is specified). Example unit types are: SYSDA, TAPE, 3380, 3390.

No default unit type exists. You cannot specify different device types for the two units used in dual logging of a tracker log data set type.

PRICOUNT

Specifies the number of devices to be allocated for the data set or primary data set. This is most useful for tape devices because multiple tapes can be mounted, thus eliminating the need to wait for tape rewind and new tape mount.

The default for PRICOUNT is 0 (zero). Specifying 0 results in data sets consisting of up to five volumes for tape data sets. Any other value for PRICOUNT results in a maximum of 255 volumes per data set.

SECUNIT

Specifies the unit type to be used for the secondary data set. Example unit types are: SYSDA, TAPE, 3380, 3390.

No default unit type exists. You cannot specify different device types for the two units used in dual logging of a tracker log data set type.

SECCOUNT

Specifies the number of devices to be allocated for the secondary data set. This is most useful for tape devices, because multiple tapes can be mounted, thus eliminating the need to wait for tape rewind and new tape mount.

The default for SECCOUNT is 0 (zero). Specifying 0 results in data sets consisting of up to five volumes for tape data sets. Any other value for SECCOUNT results in a maximum of 255 volumes per data set.

The recommendation is to specify `PRICOUNT(0)` and `SECCOUNT(0)` to limit the number of volumes to five or less when allocating archived SLDS during restart. Attention needs to be taken to ensure that the amount of data being archived will fit into five volumes if tape data sets are being used.

If you are using database readiness level on your tracking subsystem, the recommendation is that the UNIT be a direct access device, because the volume mounting characteristics of tape devices can result in unsatisfactory operation.

SPACE

Specifies the space allocation, in megabytes, for each SLDS data set.

The range of valid values is 1 to 4095. No default space allocation exists.

BLOCK

Specifies the block size to be used for all SLDS data sets. No default block size exists.

DFSRSRxx

This value should be a fairly large number to ensure good RSR performance. A block size of 32 KB is recommended for tape archiving and half-track for DASD (22KB for 3380 and 26 KB for 3390 devices). The block size must be at least as large as the largest incoming IMS log record. The range of valid values is 4096 to 32760.

MODE

Specifies whether SINGLE or DUAL SLDS logging is required. The default is SINGLE.

TMINAME

Specifies the one- to four-character transport manager instance name that is used to identify a particular transport manager subsystem in the RSR complex.

If specified, TMINAME overrides the TMINAME= parameter in the IMSCTRL macro.

If no *tminame* is specified, IMS will use a series of blanks as the default *tminame*, and associates IMS with the TMS that uses a default instance name.

TRKMODS

Specifies the tracking subsystem's MODSTAT DFSMDA member names for tracking MODSTAT status of the active subsystem.

This parameter is only valid for active IMS subsystems, and use of the parameter prohibits sharing of DFSRSRxx members between active subsystems.

The DD names specified must be included in the dynamic allocation list at the tracking site, and the data sets defined by these DD names need to be available to the tracking subsystem.

A tracking subsystem that was previously an active subsystem is not required to use the same DD name as was specified for it when it was an active subsystem.

MODSTAT

Specifies the DFSMDA member name used to track the MODSTAT data set.

This parameter is only valid for IMS active subsystems.

No default member name exists. If MODSTAT is not specified, no MODSTAT tracking is done.

MODSTAT2

Specifies the DFSMDA member name used to track the MODSTAT2 data set.

This parameter is only valid for XRF alternate subsystems.

No default member name exists. If MODSTAT2 is not specified, no MODSTAT2 tracking occurs.

Member DFSSPMxx

This member allows you to override the default buffer definitions for the storage pools managed by the DFSPPOOL storage manager.

Default storage pool definitions are generated by IMS. Each definition contains information on the pool including the default buffer definitions.

The buffer size definitions contain the buffer sizes the Storage Manager is allowed to choose from when allocating a buffer from the pool. For each buffer size, the definition specifies how many buffers to obtain in the primary block, how many buffers to obtain in secondary blocks, and whether or not to obtain the primary block when the pool is allocated. If the primary block is obtained during initialization, it is not released during compression.

The buffer definitions shown are the existing default definitions.

For all pools, 16 bytes are added to the buffer sizes for prefix and suffix information. For CIOP, HIOP, EMHB, LUMP, LUMC, and SPAP pools, an additional eight bytes are added for the overlay detection constant.

For EMHB, the buffer size includes the prefix and data portion. For information on the prefix length, see the EMHBHL field of the expedited message handler block control block that is mapped by the DBFEMHB macro.

DFSSPMxx

AOIP

	BUF1	BUF2	BUF3	BUF4	BUF5	BUF6	BUF7	BUF8
SIZE	48	132	256	576	1048	2096	4192	32768
PRIM. ALLOC.	50	500	100	32	16	8	4	4
SEC. ALLOC.	50	1500	100	32	8	4	2	2
INIT.	Y	N	N	N	N	N	N	N

CIOP

	BUF1	BUF2	BUF3	BUF4	BUF5	BUF6	BUF7	BUF8
SIZE	256	512	1024	2048	4096	8192	16384	32768
PRIM. ALLOC.	64	32	32	32	16	8	4	4
SEC. ALLOC.	32	16	16	16	8	4	2	2
INIT.	Y	N	N	N	N	N	N	N

HIOP, EMHB

	BUF1	BUF2	BUF3	BUF4	BUF5	BUF6	BUF7	BUF8
SIZE	256	512	1024	2048	4096	8192	16384	32768
PRIM. ALLOC.	64	64	32	32	16	8	4	4
SEC. ALLOC.	32	32	16	16	8	4	2	2
INIT.	N	N	N	N	N	N	N	N

CESS

	BUF1	BUF2	BUF3	BUF4	BUF5	BUF6	BUF7	BUF8
SIZE	128	256	512	1024	2048	4096	8192	16384
PRIM. ALLOC.	32	32	32	32	32	16	8	4
SEC. ALLOC.	16	16	16	16	16	8	4	2
INIT.	Y	N	N	N	N	N	N	N

FPWP

	BUF	BUF	BUF	BUF4	BUF5	BUF6	BUF7	BUF8
SIZE	25	51	102	2048	4096	8192	16384	32768
PRIM. ALLOC	6	6	3	32	16	8	4	4
SEC. ALLOC.	3	3	1	16	8	4	2	2
INIT.				N	N	N	N	N

LUMP, LUMC

	BUF1	BUF2	BUF3	BUF4	BUF5	BUF6	BUF7	BUF8
SIZE	128	256	512	1024	2048	3064	4096	33024
PRIM. ALLOC.	32	32	32	32	16	12	8	4
SEC. ALLOC.	32	16	16	16	8	12	8	2
INIT.	N	N	N	N	N	N	N	N

Figure 11. Default Buffer Definitions

With DFSSPMxx, you can override buffer definitions for any of the pools identified in Figure 11:

- AOIP
- CESS
- CIOP
- EMHB
- FPWP
- HIOP

LUMC

LUMP

Exception: If you establish an FDBR region, all pool names that you specify are ignored. The FDBR region internally specifies the following values for all pools regardless of whether they are explicitly specified by the control statements or defined by SYSGEN as default:

- 2 Number of buffers in the primary storage allocation
- 2 Number of buffers in secondary storage allocation
- N Primary storage allocation during IMS initialization =

You can define a maximum of thirty-two different buffer sizes that can be used to satisfy requests for storage. The size of the buffer that is used to satisfy a request is determined on a “best-fit” basis.

The FPL= statement is used to specify buffer definition overrides. Multiple buffer definitions are allowed on a single statement. You can have one or more FPL= statements within the DFSSPMxx member. If the member contains more than thirty-two buffer definitions for a single pool, only the first thirty-two are used. Remaining definitions are ignored.

DFSSPMxx is not required to initialize IMS. It is intended only to override existing IMS storage pool definitions; therefore, no default member is supplied.

The format of the buffer definition override statement is as follows:



FPL=

Is the required keyword for IMS storage pool buffer override definitions. This keyword must be in position 1. The FPL= statement cannot be continued; however, more than one statement is allowed per pool.

poolname

Is the four-character name of the IMS storage pool. The valid names are:

AOIP
CESS
CIOP
EMHB
FPWP
HIOP
LUMC
LUMP

size

Specifies the buffer size, in bytes. The value must be a one- to five-digit number between 8 and 65536. The buffer size is rounded up to the next multiple of 8. If the rounded value is not unique, it is discarded. You can specify values in 1K increments; that is, 1K, 2K, 3K, up to 64K.

pbuf

Specifies the number of buffers in the primary storage allocation. The value must be a one- to five-digit number between 2 and 65535. If the size of the primary allocation exceeds the upper expansion limit, the default is 2.

sbuf

Specifies the number of buffers in secondary storage allocations. The value must be a one- to five-digit number between 2 and 65535. If the size of the primary allocation exceeds the upper expansion limit, the default is 2.

init

Specifies whether (Y) or not (N) the primary storage allocation is obtained during IMS initialization. If the primary storage allocation is not obtained during initialization, it is not obtained until it is necessary to satisfy a buffer request. If an upper expansion limit is specified for the pool, and the size of either the primary or secondary storage allocation exceeds the upper expansion limit, the upper expansion limit is set to the default value of 2G-1.

If an FPL= is incorrectly specified, message DFS0639W is issued. One message is issued for each occurrence of an invalid parameter.

Related Reading: See *IMS/ESA Messages and Codes* for more information on this message.

Examples of DFSSPMxx Statement

The following are examples of how the DFSSPMxx PROCLIB member can be used to override the default buffer definitions for the CIOP pool.

```
FPL=CIOP, (248,20,15,Y), (500,20,10,N), (1016,15,8,Y)
FPL=CIOP, (2040,15,8,N), (4088,10,5,N), (8184,10,5,N)
FPL=CIOP, (16K,8,4,N), (32K,4,2,N)
```

The buffer sizes being defined range from 248 to 32768 (32K) bytes. Each buffer size is rounded up to the next multiple of 8 and then 8 bytes are added for internal processing. Table 24 shows the values used by the Storage Manager when allocating CIOP storage if the above definition is specified in the DFSSPMxx member.

Table 24. Primary and Secondary Buffer Allocations

Buffer Size	Primary Buffer Allocation	Secondary Buffer Allocation	Obtain Primary Allocation?
256	20	15	Y
512	20	10	N
1024	15	8	Y
2048	15	8	N
4096	10	5	N
8192	10	5	N
16392	8	4	N
32776	4	2	N

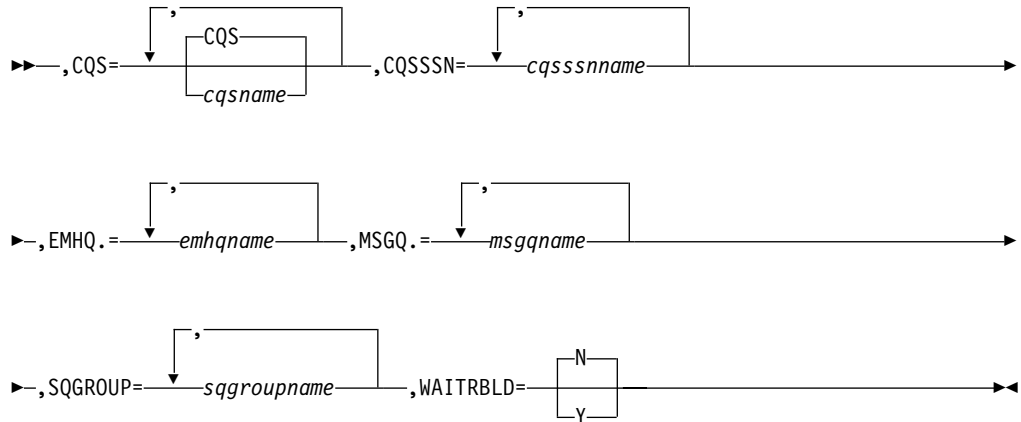
Member DFSSQxxx

The DFSSQxxx PROCLIB member specifies parameters related to the shared message queues and the CQS address space. The DFSSQxxx PROCLIB member applies to the DB/DC and DCCTL environments. You can find an example of the DFSSQxxx PROCLIB member, DFSSQ999, in library DFSSLIBA.

Control Statements

A DFSSQxxx member consists of one or more 80-character records. Keyword parameters are coded in positions 1-71 in free format and can contain leading and trailing blanks. Positions 72-80 are ignored. Multiple keywords can be specified in each record; commas are used to delimit keywords.

Errors encountered during DFSSQxxx PROCLIB member processing cause IMS initialization to terminate.



CQS=

Specifies the one- to eight-character name of the PROCLIB member that contains one of the following:

- The procedure for the CQS address space.

When shared message queues are requested the IMS control region automatically starts the CQS procedure during IMS initialization. The default procedure name is CQS.

- The START command.

The START command begins in column 1 with the characters START. The START command and its parameters must not extend beyond column 71.

When shared message queues are requested, the IMS control region issues the user-specified START command to start the CQS address space.

CQSSSN=

Specifies the one-to four- character subsystem name of the CQS address space. IMS uses this name to connect to the proper CQS address space. When connecting IMS to CQS, you must specify the same value on CQSSSN= and on the SSN= parameter of the CQSIPxxx PROCLIB member for the target CQS. The parameter is required and no default exists.

EMHQ=

Specifies the 1- to 16- character name of the primary structure that contains the shared expedited message handler queues. If IMS is Fast Path capable, the shared expedited message handler queues are required and you must specify this parameter. If you specify this parameter for an IMS which is not Fast Path capable, the parameter is ignored.

If you specify an EMHQ= structure, you must also specify shared message queues on the MSGQ= parameter. IMS requires the message queues to

DFSSQxxx

process system messages that result from Fast Path activity. If the EMH queues are built in a coupling facility structure, the message queues must also be in a structure.

The name you specify for EMHQ= must not be the same as the name you specify for MSGQ= The name must also be specified to CQS on the STRNAME= parameter in the CQSSLxxx and CQSSGxxx PROCLIB members.

Related Reading: For additional information on the structure name, see *IMS/ESA Common Queue Server Guide and Reference*.

MSGQ=

Specifies the 1- to 16- character name of the primary structure that contains the shared message queues. If the message queues are built in a coupling facility structure, the EMH queues, if used, must also be in a structure.

The name you specify for MSGQ= must not be the same as the name you specify for EMHQ= The name must also be specified to CQS on the STRNAME= parameter in the CQSSLxxx and CQSSGxxx PROCLIB members.

Related Reading: For additional information on the structure name, see *IMS/ESA Common Queue Server Guide and Reference*.

SQGROUP=

Specifies a one- to five- character identifier. IMS concatenates this identifier to DFS to create the XCF IMS shared queues group name. You must specify the same identifiers for all IMS subsystems that share the same set of structures. You can use the same identifier for the CQSGROUP= parameter in the CQSIPxxx PROCLIB member. This parameter is required and there is no default.

Related Reading: For information on the CQSGROUP= parameter, see *IMS/ESA Common Queue Server Guide and Reference*.

WAITRBLD=Y|N

Specifies whether activity against the EMHQ structure waits while the structure is being rebuilt. If you specify Y (yes), all activity against the EMHQ structure waits until the structure rebuild completes. If you specify N (no), activity against the EMHQ structure continues while CQS rebuilds the structure.

The value specified for this keyword remains in effect for the life of the structure and cannot be changed.

Restriction: This parameter does not apply to the MSGQ. The MSGQ must wait for rebuild to complete before activity can resume.

If you change the WAITRBLD=parameter, you must first deallocate the fast path structure.

If errors are encountered during DFSSQxxx PROCLIB member processing (for example, no MSGQ= parameter is specified), IMS initialization terminates.

Member DFSVFLqq

This member is not applicable in the DBCTL environment.

Using Virtual Fetch

MPP dependent regions can use MVS Virtual Fetch. To use the IMS support for virtual fetch, complete the following steps:

1. Use the IMS.PROCLIB data set allocated prior to stage 2 of IMS system definition and the MVS utility IEBUPDTE to create IMS.PROCLIB members that identify modules to be managed by virtual fetch.

Related Reading: See *MVS/ESA Data Administration: Utilities* for information about the MVS IEBUPDTE utility. Member names must be of the form DFSVFLqq, where qq is a two-character suffix. The record format is:

Position	Contents
1-71	Module names, separated by a comma
72-80	Ignored

The first name on a record can have leading blanks. The last name on a record is followed by a comma and one or more blanks. The last name on the last record is followed only by one or more blanks.

You can specify as many module names as you want. If a module (that is, an application program in an MPP) that is to be managed by virtual fetch has ALIAS names that would ordinarily be invoked via MVS LINK, you should also specify those ALIAS names.

2. Link-edit the application program modules using the IMS reentrant DL/I language interface. You must specify the true attributes. For example, if a module is reusable only, do not specify "RENT."

The IMS DL/I language interface is not reentrant. Any IMS application programs that were designed to be reentrant or serially reusable can use virtual fetch only after being link-edited with the IMS language interface.

3. In the MPP step execution JCL:
 - Ensure that there is a DD statement for IMS.PROCLIB with the ddname PROCLIB.
 - Use the EXEC statement parameter VSFX to specify the designated PROCLIB member. VSFX is a two-character suffix for the PROCLIB member that contains the names of the application programs that are not preloaded and that are to be managed by virtual fetch.

If you specify an invalid member name, or if the record format is incorrect, IMS ignores the names and issues a message; however, initialization continues. The member name must be unique.
 - If the virtual storage obtained is to be freed, ensure that you specify the parameter VFREE on the EXEC statement as Y.

For a description of the EXEC parameters, see "DFSMPR Procedure" on page 211.

Member DFSVSMxx

The control statements in this member are processed during IMS initialization. Seven types of control statements are allowed:

1. The subpool definition statements, VSRBF, IOBF, DEDB, and DBD, are used to define the size and number of buffers in a subpool.
2. The shared resource pool definition statements, RESVPOOL and POOLID, control the number and use of local shared resource pools.
3. The SBONLINE statement, is used to specify sequential buffering in an online IMS or DBCTL system.

DFSVSMxx

4. The OPTIONS, DLITRACE, and OSAMOP statements allow you to influence some of the performance and debugging facilities of IMS.
5. The OLDSDEF, WADSDEF, and ARCHDEF statements are used to define the initial DASD logging environment.
6. The NODYNALLOC statement is used to disable dynamic allocation for batch IMS.
7. The CFNAMES control statement describes coupling facility structure names for Sysplex data sharing.

In an IMS batch system, these control statements are put in a data set with ddname DFSVSAMP. The data set must contain blocked or unblocked 80-character records.

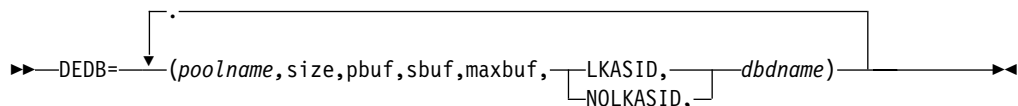
Recommendation: Do not place DFSVSAMP in the same PDS as your user application files. This causes batch jobs to abend with ABEND0C4 in DFSRTM00.

In an IMS online system, these control statements are put in the IMS.PROCLIB data set in member DFSVSMxx. The VSPEC symbolic parameter in the IMS, DBC, and DCC procedures is used to specify the suffix (xx) of the DFSVSMxx member to be used. During stage 2, IMS creates a default member with suffix 00. If a suffix is not specified on the startup procedure, the default member is used. The default values are also used if the control statements in the startup procedure are missing or invalid.

Defining Fast Path DEDB Buffer Pools

This section describes the control statements you use to define Fast Path DEDB buffer pools.

Control statements: The format of the Fast Path buffer pool is:



DEDB=

Is the required keyword for defining Fast Path buffer pools. It must be in the first position of the control statement.

poolname

The one- to eight-character name for the pool. The poolname is used as identification on display terminals and reports.

size

A three- to five-digit number specifying the size of the pool. All of the standard DEDB supported buffer sizes are supported. You can express sizes numerically, or as standard CI sizes, 1K, 2K, 4K.

pbuf

The primary buffer allocation. Specify a value from 1-9999.

sbuf

The secondary buffer allocation. Specify a value from 1-9999. This secondary allocation is used when the primary allocation runs low.

maxbuf

The maximum number of buffers allowed for this pool. It is a combination of PBUF plus some iteration of SBUF.

LKASID | NOLKASID

Indicates whether this pool is to be used as a local cache with buffer lookaside capability.

dbdname

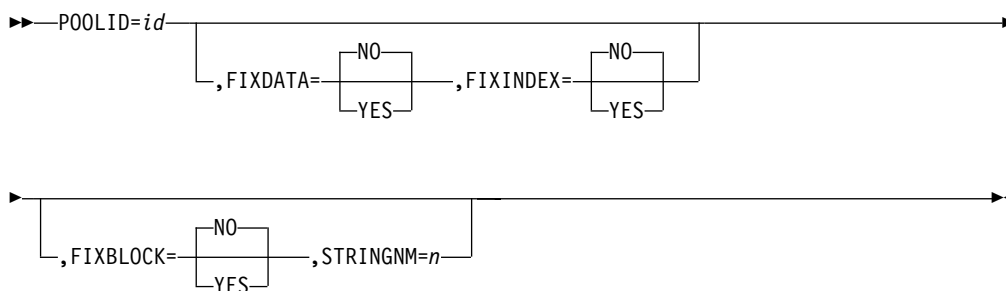
Associates a pool to a specific area or DBD. If dbdname is an area name, then the pool is used only by that area. If dbdname is a DBD name, then the pool will be used by all the areas within that DBD.

Defining VSAM Buffer Pools

Defining One or More VSAM Local Shared Resource Pools: This section describes control statements used to define multiple VSAM local shared resource pools.

Related Reading: For overview information, refer to the *IMS/ESA Release Planning Guide*.

One or more VSAM local shared resource pools can be defined through the subpool definition statement POOLID. One POOLID statement is specified for each shared resource pool. The POOLID statement must be followed by one or more VSRBF subpool definition statements, which define the subpools within that shared pool.

Control statements:**POOLID=**

Is the required keyword for VSAM shared resource pool definition. It must begin in the first position of the control statement. Only one set of subparameters can appear on each control statement. Each statement defines one shared resource pool. a given data set to a specific shared pool.

POOLIDs 0 and 1-15 can be used. The total number of POOLIDs must not exceed 16. Also refer to the paragraph under *RESVPOOL=*.

id Is a one- to four-character alphanumeric field that specifies a user identifier assigned to a shared resource pool. This parameter is required. It is used with the DBD statement to direct a given data set to a specific shared pool.

FIXDATA=YES | NO

Is an optional parameter that specifies the data shared resource pool long-term-page-fixing option. If YES is specified, all buffers in the data shared resource pool are long-term-page-fixed at initialization of the shared resource pool. If NO is specified, no buffers in the data shared resource pool are long-term-page-fixed. If this parameter is omitted, the default is NO.

FIXINDEX=YES|NO

Is an optional parameter that specifies the index shared resource pool long-term-page-fixing option. If YES is specified, all buffers in the index shared resource pool are long-term-page-fixed at initialization of the shared resource pool. If NO is specified, no buffers in the index shared resource pool are long-term-page-fixed. If this parameter is omitted, the default is NO.

FIXBLOCK=YES|NO

Is an optional parameter that specifies the I/O-related control blocks long-term-page-fixing option. If YES is specified, all I/O-related control blocks are long-term-page-fixed at initialization of the shared resource pool. If NO is specified, no I/O-related control blocks are long-term-page-fixed. If this parameter is omitted, the default is NO.

STRINGNM=n

Is an optional parameter that specifies the maximum number of VSAM I/O requests that can be concurrently active. It can be used to override the MAXREGN parameter specified at system generation time and the PST value on the EXEC statement. The specified value must be a decimal number between 1 and 255. The default is 255. The value specified should be a number as close as possible to the maximum number of regions expected to be running concurrently, including those that could be dynamically started. If the value is omitted or invalid, the default is the MAXREGN value or the PST value, which, if present, overrides the MAXREGN value.

The above buffer fix and string number parameters must be separated by commas and coded without intervening blanks. They can be coded in any order. If any parameter is invalid, the remainder of the statement is ignored and defaults apply for the remaining parameters.

The POOLID buffer fix and string number values override the VSAMFIX and STRINGMX parameters on the OPTIONS statement for that particular shared pool.



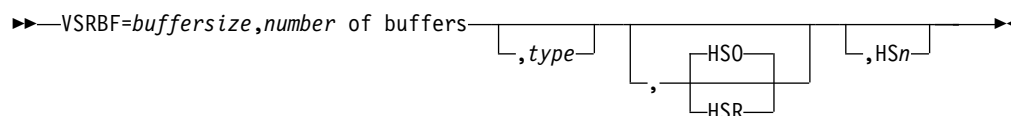
RESVPOOL=shared-pool-number

shared-pool-number represents the VSAM local shared resource pool number of the shared pool that IMS is not allowed to use. One or more numbers can be specified in any order and must be separated by commas, but without any blanks. The range is from 1 to 15. Shared resource pool 0 is reserved for IMS and is not allowed. A blank must follow the last number specified, and the remaining portion of the statement is not examined.

The RESVPOOL statement is only applicable if you have a product other than IMS using VSAM local shared resource pools that IMS is not allowed to use.

Defining VSAM Subpools

Control statements:

**VSRBF=**

Is the keyword for VSAM subpool definition. It must begin in the first position of the control statement. Only one set of subparameters can appear on each control statement. Each statement defines one subpool.

buffer size

Is a three- to five-digit number specifying the buffer size for this subpool. Acceptable values are 512, 1024, 2048, 4096, 8192, 12288, 16384, 20480, 24576, 28672, and 32768.

number-of-buffers

Is a one- to five-digit number (3 to 32767) specifying the number of buffers in this subpool. If you specify a number of buffers less than the minimum number required, IMS increases the number to the minimum and issues a warning message.

type

An optional parameter (data or index) that specifies the type of the subpool in the shared resource pool. The type parameter is a one-character field that must be either *I* for an index subpool or *D* for data subpool. If this parameter is invalid or not coded, data subpool is assumed.

If you do not specify either, both data and index occupy the same buffer subpool. When you specify *D* or *I*, you are also required to specify all data and index subpools to handle buffering requests.

The following example provides buffers for data (2K) and index (4K) if these are the only VSRBF statements in DFSVSMxx.

```
VSRBF=2048,20
VSRBF=4096,40
```

In the next example, ten index buffers (2K) are provided in the first line, and ten data buffers (2K) are provided in the second line. The third line provides 20 data buffers (4K). No index buffers of 4K are provided in this example.

```
VSRBF=2048,10,I
VSRBF=2048,10,D
VSRBF=4096,20
```

Although the index and data components of a VSAM data set must reside in the same shared resource pool, they can be assigned different subpools. You can define subpools that accommodate each component of the data set. Note that an index subpool cannot exist without a data subpool (within a given shared resource pool), but a data subpool can exist without an index subpool. If any index subpool exists in a local shared resource pool, all index components of VSAM data sets in that shared pool must share the index subpool. The CONTROLINTERVALSIZE parameter in the listing of the VSAM catalog that shows the control interval sizes assigned to the components can help you determine the buffer sizes needed for both index and data components.

HSO|HSR

Optionally specifies the kind of action IMS should take if Hiperspace

(extended storage on MVS) buffering for this subpool is not available. If this parameter is invalid or not coded when HS*n* is given, the HSO option is assumed.

HSO

Indicates that Hiperspace buffering is optional and IMS can continue initialization without Hiperspace buffering.

HSR

Indicates that Hiperspace buffering is required and IMS must terminate if Hiperspace buffering is not available.

HSO and HSR are the minimum truncations of which up to the full terms (HSOptional and HSRequired) are recognized.

HS*n*

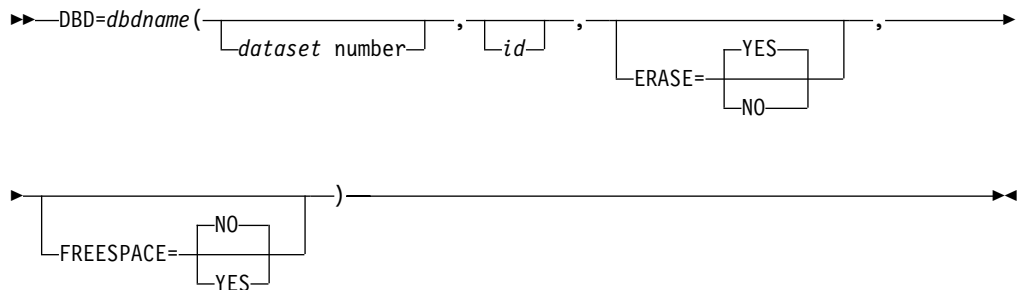
Is an optional one- to eight- digit number *n* ranging from (3 to 16777215) that specifies the number of Hiperspace buffers to build for this subpool.

HSO|HSR and HS*n* are valid only on 4K and larger boundaries. See the following example.

```
4096,10,HS40,HSR
8192,10,HS50,HSRE
12288,4,HS12
16384,4,HS20,HSO
20480,4,HSOP,HS8
24576,4,HSOPTIONAL,HS8
28672,1,HS9,HSREQUIRED
32768,1,HSREQ,HS9
```

Only the buffer size and number-of-buffers parameters are positional. The type and HS parameters can be coded in any order but must be separated by commas without any intervening blanks. If any parameter is invalid, the remainder of the statement is ignored and defaults apply for the remaining parameters.

Using the VSRBF statement without any POOLID statement can still define one, and only one, VSAM local shared resource pool. However, even with one VSAM shared resource pool, you can now use the TYPE parameter to define either an index or data subpool within the shared resource pool.



DBD=

If coded, specifies that the data set having a matching ID parameter, as defined on the POOLID shared pool definition statement, is to be directed to the indicated shared resource pool.

A shared resource pool is assigned to a data set based upon the shared resource pool identifier specified on the DBD statement for that data set. Then a subpool within the assigned shared resource pool is assigned to the data set based upon buffer length.

If no subpool in the shared resource pool is big enough, an attempt to open the data set fails. An open is then retried in the default shared pool, which is the first shared resource pool you define. To avoid another open failure, subpools of the largest sizes that might be encountered in the system should be defined in the default pool.

If no DBD statement is coded for a data set, the default shared resource pool is assigned.

dbdname

Specifies the name on the DBD macro statement NAME= keyword.

Related Reading: See “Database Description (DBD) Generation” in *IMS/ESA Utilities Reference: Database Manager*, for information on coding the DBD macro statement.

data set number

Identifies the specific data set of a data set group within a database (identified by the dbdname parameter) that requests assignment of a specific shared pool. The number is an IMS internally assigned value between 1 and 10.

For data organizations such as primary index, unique secondary index, and HISAM without dependent segments, the primary data set of the data set group is assigned data set number 1. No secondary data set of the data set group exists for these data organizations.

For data organizations such as non-unique secondary index and HISAM with dependent segments, the primary data set of the data set group is assigned data set number 1, and the secondary data set of the data set group is assigned data set number 2.

For hierarchic direct data organization, the data set group always consists of a single data set. The data set of the first data set group is assigned data set number 1, and data set numbers for subsequent data set groups are sequentially assigned numbers 2,3, ...10. The maximum number of data set groups for a database is 10.

id Specifies the user-defined identifier that is assigned to a specific shared resource pool. The ID is a one- to four-character alphanumeric field and must be equal to the ID assigned to the specific shared resource pool on the POOLID statement.

ERASE=YES|NO

Indicates the treatment of logical records that are deleted. YES indicates that the deleted record should be erased. NO indicates that the deleted record should not be erased but should be marked as a deleted record. The default is YES.

FREESPACE=YES|NO

Indicates the treatment of the defined free space percent in the KSDS. If

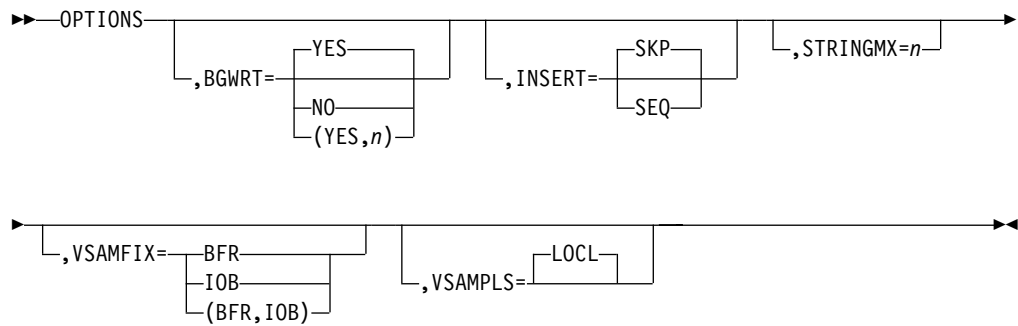
YES is specified, the defined free space should be preserved. If NO is specified, the defined free space should not be preserved. The default is NO.

Defining VSAM Performance Options

This section describes the OPTIONS control statement parameters that can be used to influence the performance of the IMS DL/I buffering services.

The word OPTIONS starting in position one identifies the OPTIONS statement. You can specify the parameters in any order, but you must separate them with commas. A blank must follow the last parameter. The remaining portion of the statement is not examined. You cannot continue an OPTIONS statement on a subsequent line, but you can provide several statements. If an OPTIONS parameter appears more than once, its setting is determined by the last occurrence. The OPTIONS statement and all of its parameters are optional.

Control Statements:



BGWRT=

Specifies whether (YES) or not (NO) the background write (BGWRT) function of the buffer handler is to be active. You can also activate background write by coding BGWRT=(YES,*n*) or omitting the parameter. *n*: is a two-digit number from 10 to 99 specifying the percentage of buffers in each subpool to be considered as candidates for writing by the background write function; the default is 34.

Related Reading: For an explanation of background write, see “Determining Which VSAM Options You’ll Use” in *IMS/ESA Administration Guide: Database Manager*.

INSERT=

Specifies the insert mode that the buffer handler uses when inserting new KSDS logical records into any of the following databases: HISAM, Single segment HISAM, and INDEX.

If a program inserts many new root segments in sequence by key, specifying INSERT=SEQ causes the buffer handler to use VSAM sequential mode PUTs. VSAM leaves free space (if you specify it in the DEFINE) for VSAM data sets in CIs created for the new records that are inserted using VSAM’s mass insert. Specifying INSERT=SKP, or omitting the parameter, causes the buffer handler to use VSAM skip sequential mode PUTs if the database is in sequential mode, or to use direct mode PUTs if the database is not in sequential mode. For more information on sequential mode databases, see “VSAM Subpool Definition” on page 379

page 379. In either mode, VSAM does not leave free space, even if it is specified in the DEFINE regardless of whether DL/I inserts in load or insert mode.

The default is SKP, except in utilities, where it is SEQ.

You can override INSERT=SEQ with the FREESPACE parameter of the DBD control statement. FREESPACE=YES causes the buffer handler to use VSAM sequential mode PUTs. Specifying FREESPACE=NO, or omitting the parameter, causes the buffer handler to use VSAM skip sequential mode PUTs if the database is in sequential mode, or to use direct mode PUTs if the database is not in sequential mode.

STRINGMX=

Specifies the maximum number of VSAM requests that can be concurrently active. *n* is any number between 1 and 255. If you omit this keyword, the default value is the value specified on either the MAXREGN= parameter on the IMSCTRL macro or the PST= parameter on the IMS procedure EXEC statement (which overrides the MAXREGN= value). The STRINGMX= keyword overrides either of these two values.

VSAMFIX=

Causes VSAM resource pools to be page fixed in main storage.

BFR

Specifies that the buffer subpools and control blocks should be fixed. If you define a large number of buffers and then use VSAMFIX=BFR the system might abend.

IOB

Specifies that the input/output related blocks should be fixed.

Either or both operands can be specified. The syntax for specifying both operands is (BFR,IOB) or (IOB,BFR).

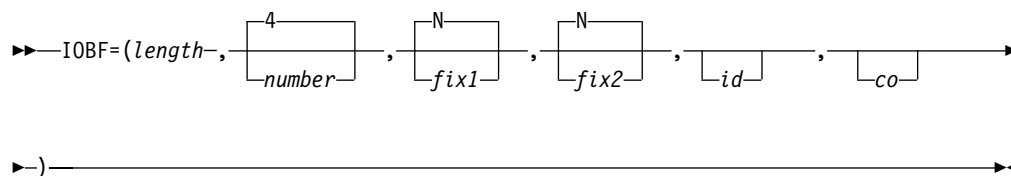
VSAMPLS=LOCL

Specifies that the VSAM shared resource pools are to be built. The only valid value is the default, LOCL.

Defining OSAM Buffer Pools

Defining OSAM Subpools

Control statements:



IOBF=

Is the required keyword for OSAM subpool definition. It must begin in the first position of the control statement. Only one set of subparameters can appear on each control statement.

length

Specifies the length of the buffers in the subpool. Specify the value in bytes, between 512 and 32000. Depending on the value that you specify, IMS rounds it up to 512, 1024, 2048, and thereafter to multiples of 2048. You can code specifications of 1024 and above as 1K, 2K, 4K, and thereafter round them up to multiples of 2K to a maximum of 32K. If you are defining a subpool for a BISAM data set, the buffer size must be at least the block size plus 16 bytes. If you enter an invalid value, IMS ignores the entire IOBF statement. This parameter is required.

number

Specifies the number of buffers in the subpool. If specified, you must choose a value between 4 and 32767. If you choose a number greater than 32767, the default value of 255 is used. If not specified, the default value is 4. If this parameter is invalid, the remainder of the entry is ignored, and defaults apply for all remaining parameters. This parameter is optional.

fix1

Specifies the buffer long-term page-fixing option. If you specify Y, all buffers and buffer prefixes associated with this subpool are long-term page-fixed at initialization of the subpool. If you specify N, no buffers associated with this subpool are long-term page-fixed at initialization of the subpool. The default is N. This parameter is optional.

fix2

Specifies the buffer prefix long-term-page-fixing option. If you specify Y, all buffer prefixes associated with this subpool and the subpool header are long-term-page-fixed at initialization of the subpool. If you specify N, the subpool header and all buffer prefixes associated with this subpool are not long-term-page-fixed at initialization of the subpool. The default is N. This parameter is optional.

id

Specifies a user-defined identifier to be assigned to a subpool. This ID is a one- to four-character alphanumeric field and is used with the DBD statement to assign a specific subpool to a given data set. If this parameter is not coded, IMS assigns a null ID to the subpool.

If two or more subpool definition statements specify the same buffer size (without the subpool ID), the number of buffers from the statements are summed. If the total does not exceed 32767, IMS builds a single subpool with the total number of buffers; however, if it does exceed 32767, a single subpool with 255 buffers is built.

co

Specifies the subpool caching option. For more information on OSAM data caching, see

Related Reading: You can specify the caching option in one of the following ways: see “OSAM Subpool Definition” on page 379.

omitted

No data caching.

Caching is not active for the subpool.

N

No data caching.

Caching is not active for the subpool.

A

Cache all data.

Write all data read from DASD and all changed data to the coupling facility

If you are defining a subpool for a BISAM data set, be sure that the buffer size is at least the block size plus 16 bytes.

►►—OSAMOP—IOSCB=NO—◄◄

The OSAMOP statement allows options specific to the OSAM access method.

IOSCB=NO

Indicates that no OSAM resources are to be allocated to the batch applications that do not require OSAM services.

►►—OSAMOP—OSAMGTF=YES—◄◄

OSAMGTF=YES

Enables OSAM to generate GTF trace records while in batch.

Specifying Sequential Buffering for an Online System

Use the SBONLINE control statement to request sequential buffering for an IMS or CICS system. This statement applies to the IMS DB/DC or DBCTL environments. SBONLINE requests loading of SB modules and specifies the maximum amount of buffer space that SB can use.

The format of the SBONLINE control statement is as follows:

Control statements:

►►—SBONLINE—
 └─,MAXSB=nnnn—┘◄◄

SBONLINE

Specifies that sequential buffering modules are to be loaded in the IMS DB/DC or DBCTL environments, thereby allowing subsequent use of sequential buffering.

By default, IMS does not load the SB modules. This prevents an increase in virtual storage requirements.

MAXSB=nnnn

Specifies the maximum amount of space (in kilobytes) that can be allocated to SB buffers. This number represents the total amount of space to be allocated for **all** concurrently executing programs, not for just one individual program. Setting a limit on the amount of SB space protects you from allocating excessive virtual storage. The default for MAXSB is unlimited space.

When the MAXSB limit is reached, IMS stops allocating SB buffers until terminating programs release buffer storage.

Defining Serviceability and Trace Options

This section describes the OPTIONS control statement parameters, which can be used to influence the serviceability of IMS. For example, the OPTIONS statement allows you to specify which IMS traces should be turned on automatically by the system during IMS initialization.

The word **OPTIONS** starting in position 1 identifies the **OPTIONS** statement. You can specify the parameters in any order, but you must separate them with commas. A blank must follow the last parameter. The remaining portion of the statement is not examined. You cannot continue an **OPTIONS** statement on a subsequent statement, but you can provide several statements. If an **OPTIONS** parameter appears more than once, its setting is determined by the last occurrence. The **OPTIONS** statement and all of its parameters are optional.

Recommendation: Select ON for the following parameters:

- DL/I=
- LOCK=

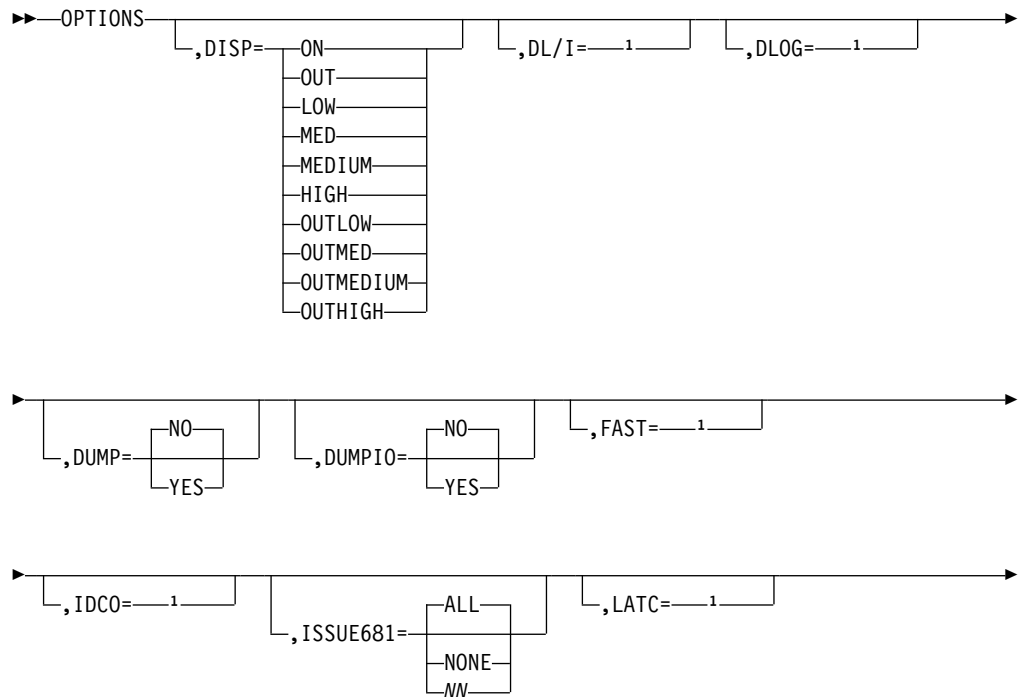
For an online system, select ON for these additional parameters:

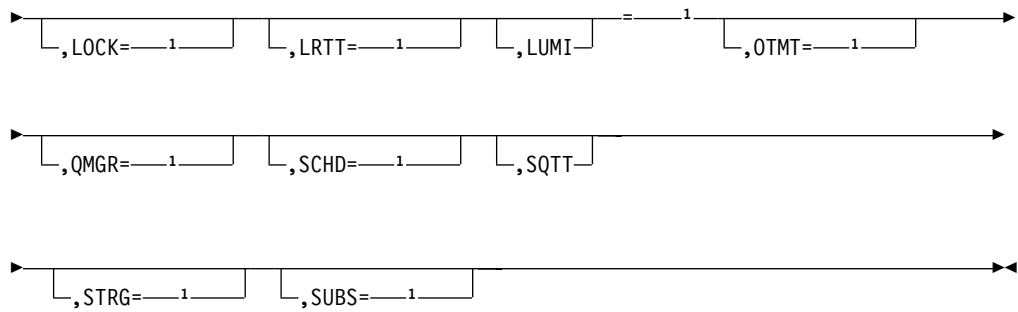
- DISP=
- SCHD=

Turning on the trace tables specified by these parameters does not cause a noticeable performance impact, and each of the trace tables can be extremely helpful to you in diagnosing a variety of problems that might occur in your environment. For certain types of problems, the IMS Support Center needs the trace output. If it's not available, you may need to re-create the problem, which will take time and delay the problem resolution.

Related Reading: For more information about all of these traces, see *IMS/ESA Diagnosis Guide and Reference*.

Control statements:





Notes:

1 This parameter accepts values from the list of common options.

Note: 1. This control statement parameter accepts values from the list of common options. You are required to choose an option if you use a particular parameter (for example, see DISP=)

Common Options:

ON

Turns on the trace table.

OUT

Initializes and turns on the trace table for output to the log. Log records 4096 bytes long are written to the log when trace records are written.

LOW

Turns on the queue manager trace for all low volume events.

MED

Turns on the queue manager trace for all low and medium volume events.

MEDIUM

Turns on the queue manager trace for all low and medium volume events.

HIGH

Turns on the queue manager trace for all low, medium, and high volume events.

OUTLOW

Initializes the trace table for output to the log and turns it on for tracing all low volume events.

OUTMED

Initializes the trace table for output to the log and turns it on for tracing all low and medium volume events.

OUTMEDIUM

Initializes the trace table for output to the log and turns it on for tracing all low and medium volume events.

OUTHIGH

Initializes the trace table for output to the log and turns it on for tracing all low, medium, and high volume events.

DISP=

Activates an IMS trace and traces the calls of the IMS dispatcher. No default

value exists. If this parameter is omitted, the trace table is not turned on by the OPTIONS statement. It can be turned on and off by means of the online /TRACE command.

DISP= is applicable to the online system only.

For a list of the valid options for this parameter, see “Common Options” on page 352.

DL/I=

Activates an IMS trace that traces the calls of certain DL/I modules and the functions invoked by these DL/I modules. No default value exists. If this parameter is omitted, the trace table is not turned on by the OPTIONS statement. It can be turned on and off by the online /TRACE command.

The DL/I trace, the program isolation trace, and the lock trace (described below) share the same trace table.

This parameter can also be used to turn on the DL/I retrieve trace table, but it does not control the number of table entries, which is fixed at 255.

For a list of the valid options for this parameter, see “Common Options” on page 352.

DLOG=

Activates an IMS trace. The DASD log trace traces the activity of the IMS DASD logger. No default value exists. If this parameter is omitted, the trace table is not turned on by the OPTIONS statement. It can be turned on and off by the /TRACE command.

For a list of the valid options for this parameter, see “Common Options” on page 352.

DUMP=

Provides a serviceability aid. If you specify NO or omit the parameter and an abnormal condition is encountered, the buffer handler issues a pseudoabend. If you specify YES and an abnormal condition is encountered, the buffer handler issues a standard abend, which results in the control region abnormally terminating also. If DUMP=YES and any abnormal termination occurs, the modified or newly created buffers in the pool at that time are not written to the data set.

DUMPIO=

Provides a serviceability aid for analyzing OSAM I/O errors. If you specify YES and an OSAM I/O error occurs, the IMS region abnormally terminates with a U0764. If the DUMPIO option is requested, the DFS07621 error message is not displayed.

FAST=

Activates an IMS trace that traces DBF trace entries from various Fast Path modules. No default exists. If this parameter is omitted, the trace is not turned on by the OPTIONS statement. It can be turned on and off by the online /TRACE command.

Recommendation: Run the FPTRACE in a test environment only. FPTRACE output is very large and can impact performance.

Related Reading: For more details on the FPTRACE, see *IMS/ESA Operator's Reference*.

For a list of the valid options for this parameter, see "Common Options" on page 352.

IDC0=

Indicates that tracing of errors in modules DFSCNXA0 and DFSIDCO0 is to be activated or deactivated.

For a list of the valid options for this parameter, see "Common Options" on page 352.

ISSUE681=

Specifies the number of DFS681I messages to be issued per second within a batch or BMP region.

ALL

Allows all DFS681I messages generated to be issued. All is the default.

NONE

Allows no DFS681I messages to be issued.

NN

Is the number (1 - 99) of DFS681I messages that you allow to be issued. If some DFS681I messages are not issued, DFS683I is issued and states the number of DFS681I messages omitted.

LATC=

Activates an IMS trace. LATCH traces IMS latch activity. No default value exists. If this parameter is omitted, the trace table is not turned on by the OPTIONS statement. It can be turned on and off by the /TRACE command.

For a list of the valid options for this parameter, see "Common Options" on page 352.

LOCK=

Activates an IMS trace that traces the lock activity of certain DL/I modules. No default value exists. If this parameter is omitted, the trace table is not turned on by the OPTIONS statement. It can be turned on and off by the online /TRACE command. Turning on the lock activity trace also turns on the program isolation trace.

The lock trace, the program isolation trace, and the DL/I trace (described above) share the same trace table.

For a list of the valid options for this parameter, see "Common Options" on page 352.

LRTT=

Activates an IMS trace that traces the activity of the log router component of the RSR tracking system. No default exists. If this parameter is omitted, the trace is not turned on by the OPTIONS statement. It can be turned on and off by the online /TRACE command.

For a list of the valid options for this parameter, see "Common Options" on page 352.

LUMI=

Trace table for OCO LUM code that supports an LU 6.2 device. It contains the following information:

- Record written on entry and exit to LU 6.2 device PL/AS modules.
- Information from APPC/MVS verb with acceptable or unacceptable return code.

For a list of the valid options for this parameter, see “Common Options” on page 352.

OTMT=

Activates an IMS trace. OTMT traces the flow of control through IMS OTMA. No default exists. If this parameter is omitted, the trace is not turned on by the OPTIONS statement. It can be turned on and off by the online /TRACE command.

For a list of the valid options for this parameter, see “Common Options” on page 352.

QMGR=

Activates an IMS trace for an online system. QMGR traces calls made to the IMS queue manager. No default value exists. If you omit this parameter, the trace table is not turned on by the options statement, but can be turned on and off by means of the online /TRACE command.

For a list of the valid options for this parameter, see “Common Options” on page 352.

SCHD=

Activates an IMS trace. SCHD traces the calls of the IMS scheduler. No default value exists. If this parameter is omitted, the trace table is not turned on by the OPTIONS statement. It can be turned on and off by the online /TRACE command.

For a list of the valid options for this parameter, see “Common Options” on page 352.

SQTT=

Activates an IMS trace for an online system. SQTT traces the shared queues interface including:

- Initialization of the connection to CQS
- CQS requests
- Notification of work available from CQS
- Termination of the connection to CQS

No default value exists. If you omit this parameter, the trace table is not turned on by the OPTIONS statement, but can be turned on and off by means of the online /TRACE command.

For a list of the valid options for this parameter, see “Common Options” on page 352.

STRG=

Activates an IMS trace. STRG traces all calls to the IMS storage manager that require modifications to one of the following pools:

CESS
CIOP

EMHB
 FPWP
 HIOP
 LUMC
 LUMP

If this parameter is omitted, the trace table is not turned on by the OPTIONS statement; instead, it can be turned on and off with the online /TRACE command.

STRG= is applicable only to the online system.

For a list of the valid options for this parameter, see “Common Options” on page 352.

SUBS=

Activates an IMS trace. SUBS traces the connection and disconnection of DB2 subsystems to the IMS control region. No default value exists. If this parameter is omitted, the trace is not turned on by the OPTIONS statement. It can be turned on and off by the online /TRACE command.

For a list of the valid options for this parameter, see “Common Options” on page 352.

The LTWA= keyword is no longer used and has been eliminated from the macro format below. If LTWA= is specified, it is ignored, but it can be retained for compatibility with previous IMS releases.

The BHTRACE= keyword is no longer used and has been eliminated from the macro format below. If BHTRACE= is specified, message DFS0437I is issued, indicating an invalid parameter. This keyword has been replaced by the following keywords: DL/I=, LOCK=, DISP=, SCHD=, SUBS=, DLOG=, STRG=, IDC0=, and LUMI=. You cannot specify the number of entries; therefore, if you want to prevent trace entries from being overlaid, use the OUT option to record the entries in the log.

Also, if you wish to trace a batch application that goes to normal end-of-job, you must indicate that output is to be written to a log tape.

Table 25 is a summary of the traces that can be turned on using the OPTIONS parameter. It shows the number of trace entries and the storage requirements for the various traces.

Table 25. Trace Entries and Storage Requirements

Type	No. of Trace Tables	Entry Size	Entries Per Table	Storage Bytes Required
Latch	12	20 Hex	126	48K
Dispatcher	10	20 Hex	126	40K
Scheduler	5	10 Hex	252	20K
SubSystem	8	20 Hex	126	32K
DLog	6	20 Hex	126	24K
Fast Path	2	20 Hex	126	8K
DLI/Lock	18	20 Hex	126	72K

Table 25. Trace Entries and Storage Requirements (continued)

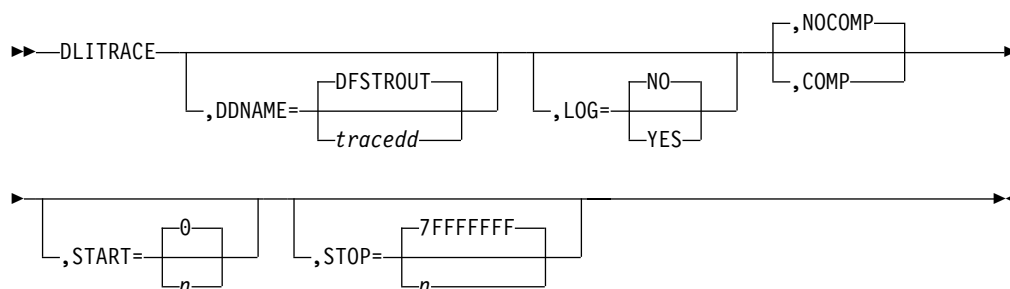
Type	No. of Trace Tables	Entry Size	Entries Per Table	Storage Bytes Required
Storage Manager	10	20 Hex	126	40K
Intercommunications	18	20 Hex	126	72K
LU 6.2	8	20 Hex	126	32K
OTMA	8	20 Hex	126	32K
Log Router	8	20 Hex	126	32K

Note: Only DLog storage is below the 16-megabyte line. For all other types listed here, storage is above the 16-megabyte line.

Related Reading: For additional information on the traces that you can define using these keywords, see *IMS/ESA Utilities Reference: System*.

Defining DL/I Call Image Trace

Control statements:



The DLITRACE statement invokes the tracing of DL/I calls in a batch environment. Specify the word DLITRACE in the first position to identify this statement. You can specify the parameters in any sequence, but you must separate them with commas; each parameter can be specified only once. If you don't specify any parameters, IMS uses the default values.

A DLITRACE statement cannot be continued on a subsequent statement. The first statement encountered in the DFSVSAMP data set establishes the options. Subsequent statements are bypassed, and message DFS2471 is issued. In cases where a parameter is repeated, the options remain set as specified by the first occurrence, and message DFS2471 is issued.

Related Reading: For details on the use of DL/I trace, see *IMS/ESA Diagnosis Guide and Reference*.

DDNAME=

Indicates that sequential data set output is requested and gives the DD statement name (tracedd) to be present in the JCL. If you omit this parameter and do not supply output DD, DDNAME=DFSTROUT is assumed, and an attempt is made to route output to the sequential data set defined by DFSTROUT DD. However, if you omit this parameter and request log output, no attempt is made to open such a sequential data set.

LOG=

Specifies whether (YES) or not (NO) DL/I call image capture output is to be routed to the IMS system log. The default is NO, and a sequential output data set will contain the trace output.

NOCOMP | COMP

Specifies whether (COMP) or not (NOCOMP) DL/I Test Program COMPARE statements are generated for both PCB comparisons and data comparisons. The default is NOCOMP.

Related Reading: For more information, see *IMS/ESA Application Programming: Database Manager*.

START=

Specifies a one- to eight-character hexadecimal value representing a count of DL/I calls issued by an application program against a specific PSB. This value indicates at which point in the program processing tracing should begin. The default value of 0 begins the tracing at the first DL/I call.

STOP=

Specifies a one- to eight-character hexadecimal value representing a count of DL/I calls. This value indicates the point at which DL/I image capture tracing against a specific PSB should stop. The default value of X'7FFFFFFF' indicates that all of the DL/I calls for the batch program are traced.

Defining DASD Logging Initialization Parameters

In a DB/DC environment, use the OLDSDEF and WADSDEF control statements to specify the log data sets and parameters to be established during DLOG initialization. These control statements are mandatory and reside within the DFSVSMxx member of IMS PROCLIB. Use the ARCHDEF control statement to modify the GENJCL.ARCHIVE command generated by IMS when it switches OLDSs.

Related Reading: For additional information on archiving OLDSs and the GENJCL.ARCHIVE command, see page 247 *When you specify ARCHDEF in the DFSVSMxx PROCLIB member, it applies only to the active system. ARCHDEF specified in the DFSRSRxx PROCLIB member is meaningful only to a remote system.*

Control Statements::

ARCHDEF Statement:

▶—ARCHDEF *aaaa*—————▶

When IMS automatically archives OLDSs in response to the ARC= parameter of the IMS procedure, it does so by internally issuing a GENJCL.ARCHIVE command to DBRC. The default command is

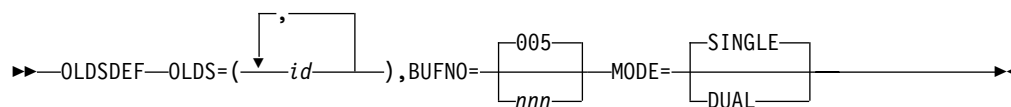
GENJCL.ARCHIVE ALL SSID (xxxx)

Use the ARCHDEF control statement to change this command to:

GENJCL.ARCHIVE *aaaa* SSID (xxxx)

In the statement above, *aaaa* represents any valid GENJCL parameter up to 63 characters in length.

OLDSDEF Statement:



The OLDSDEF statement defines the OLDS data sets to be established during initialization. It consists of one or more 80-byte records placed within the DFSVSMxx member of IMS PROCLIB. If multiple OLDSDEF statements exist, all statements are processed. The OLDSDEF syntax is free-form, and continuation statements are indicated by a character other than a blank (X'40') in column 72. Keywords are delimited from associated parameters by an equal sign (=), and keyword/parameter pairs are delimited by a comma (,).

OLDSDEF

Identifies this as an OLDSDEF control statement. This keyword must appear first.

OLDS=

Identifies a list of OLDS to be allocated at DLOG initialization, where *id* indicates a list of two-digit OLDS IDs between 00 and 99. You can specify between 3 and 100 OLDS IDs. A minimum of three OLDS IDs is required.

BUFNO=

Identifies the number of log buffers allocated for OLDS read and write operations, where *nnn* is a one- to three-digit numeric integer between 002 and 999. The default is 005.

MODE=

Identifies whether the logger is to operate in SINGLE or DUAL logging mode. The default is SINGLE.

You should provide dynamic allocation members (via the DFSMDA macro) for all OLDS data sets. All (primary and secondary) OLDS must be preallocated. You must specify the OLDS blocksize when the data set is allocated.

WADSDEF Statement:



The WADSDEF statement defines the WADS data sets to be established during initialization. It consists of one or more 80-byte records placed within the DFSVSMxx member of IMS PROCLIB. If multiple WADSDEF statements exist, only the first encountered statement is processed. The WADSDEF syntax is free-form, and continuation statements are indicated by a character other than a blank (X'40') in column 72. Keywords are delimited from associated parameters by an equal sign (=), and keyword/parameter pairs are delimited by a comma (,).

WADSDEF

Identifies this as a WADSDEF control statement. This keyword must appear first.

WADS=

Identifies a list of WADS to be allocated at DLOG initialization, where *n*

indicates a list of one-digit WADS IDs between 0 and 9. You can specify between one and ten WADS IDs. A minimum of one WADS ID is required.

You should provide dynamic allocation members (using the DFSMDA macro) for all WADS data sets. All (primary and secondary) WADS must be preallocated. The WADS block size is extracted from the DSCB. The specification of WADS duplexing is provided via the WADS= EXEC parameter specified in the control region JCL (WADS=S for single WADS, WADS=D for dual WADS).

Defining IMS Batch without Dynamic Allocation

Control statements:

►►—NODYNALLOC—◄◄

NODYNALLOC

Disables batch dynamic allocation. Message DFS2480 appears at initialization time to indicate that batch dynamic allocation is not to occur. This statement disables the default setting, which is batch dynamic allocation enabled.

Defining Coupling Facility Structure Names for Sysplex Data Sharing

If you use Sysplex data sharing, three coupling facility structure names must be passed to IMS. They are for the IRLM structure, OSAM structure, and VSAM structure.

These names are passed to IMS in the CFNAMES control statement. The purpose of this statement is to let you select which of the structures, previously defined in the MVS policy, is to be used at run time.

You can specify one or more CFNAMES control statements. Each statement can contain one or more keyword parameters. Each keyword parameter must be specified in its entirety on a single CFNAMES statement. A specific keyword parameter can not be continued from one CFNAMES statement to another. If keyword parameters are duplicated, the first parameter encountered is used and the others are registered with an error message.

The structure names that you specify in the CFNAMES control statement can be 1 to 16 characters long. All keywords in the statement must be coded, but null values for the CFOSAM and CFVSAM keywords are allowed.

IRLM: If you do not specify structure names in this PROCLIB member or specify only the IRLM structure name, the default is for the environment defaults to use two-way data sharing using the notify protocol.

If the IRLM and OSAM or VSAM structure names do not match the structure names known to the coupling facility at the time of the IDENTIFY, the IDENTIFY is rejected and IMS initialization fails. The first IMS in a data sharing group that identifies to the IRLM sets the data sharing environment for all other IMS subsystems connecting to the same IRLM structure. The first IMS, in other words, sets the structure names for the coupling facility. If, for example, the first IMS doesn't specify structure names in this PROCLIB member, no other IMS subsystems identifying to this IRLM can specify structure names (with the exception of an IRLM structure name matching the one set by the IRLM); otherwise, the IDENTIFY is rejected and IMS initialization

fails. The same thing happens if the first IMS specifies all three structure names and, later, another IMS subsystem tries to identify using one or more different structure names.

OSAM: If you are using OSAM sequential buffering, the OSAM structure registers SB buffers in the coupling facility's cross-system buffer invalidation process.

Control Statements: CFNAMES must appear in column 1. No blanks can appear between the keywords.

CFNAMES

Specifies the coupling facility structure names that must be passed to IMS.

CFIRLM=aaaaaaaaaaaaaaaa

Specifies the coupling facility lock table structure name for the IRLM.

CFVSAM=bbbbbbbbbbbbbbbb

Specifies the coupling facility XI structure name for VSAM.

CFOSAM=

You can specify either the coupling facility name only, or the coupling facility name and directory-to-element ratios.

cccccccccccccccc

Specifies the coupling facility structure name for OSAM.

(cccccccccccccccc,DIRRATIO,ELEMRATIO)

Specifies the coupling facility structure name for OSAM and the directory-to-element ratio that the CF uses to configure the structure for data caching.

The DIRRATIO and ELEMRATIO sub-parameters are optional. The installation uses the CFRM policy to specify the size of a cached structure. When the structure is allocated, its storage is subdivided to reserve space for directory entries and data elements. The directory-to-element ratio determines the proportion of the cache used for directories and data elements. The ratio is expressed as a pair of whole numbers and is used when a connection is made to the structure. The directory-to-element ratio should reflect the average number of data elements per cache entry. The size of each data element is 2 KB. OSAM data is contained in the coupling facility as multiples of 2 KB data elements. If the ratio is incorrect for an installation's use of the structure, frequent rejections of cache requests can occur because either the cache or the cache structure is full.

The result of dividing the ELEMRATIO value by the DIRRATIO value must not exceed 16, the maximum number of elements supported by an OSAM structure. If the result is greater than 16, the specified ratio is rejected and IMS supplies a ratio.

The DIRRATIO value represents the directory part of the ratio and ELEMRATIO represents the element part of the ratio. Both sub-parameters are coded as one to three digits. If you choose to specify one of the sub-parameters, you must specify both.

If you omit the DIRRATIO or specify a value of zero, IMS provides a default DIRRATIO of 1:0. If DIRRATIO specifies a non-zero value and ELEMRATIO specifies zero, IMS supplies a ratio of 1:0. This ratio causes the coupling facility to configure the OSAM structure to preclude data caching. If data

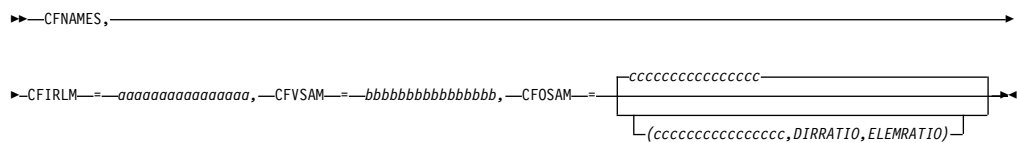
caching is precluded, the OSAM structure only supports buffer invalidation processing and does not support cache data.

If you omit the DIRRATIO and ELEMRTIO subparameters, or if they fail validation of the maximum elements number, IMS provides a default ratio.

IMS supplies a default ratio of 999:1. This ratio causes the coupling facility to configure the data caching OSAM structure to optimize buffer invalidation and not data caching. While data caching might be possible, the structure provides limited space for data elements.

Related Reading: For additional information on Sysplex data sharing, see *IMS/ESA Administration Guide: Database Manager*.

Specifies the coupling facility XI structure name for OSAM.



Using the Coupling Facility for OSAM Data Caching

The OSAM database coupling facility caching function allows you to optionally specify the caching of OSAM database buffers. For more information see the description of the caching option (co) parameter.

When you specify the "cache only changed data" option, and an application program modifies data in a subpool, changed data is written first to DASD then to the coupling facility. If you select the "cache all data" option, and an application program requests data that is not already in a subpool or in the coupling facility, that data is either read from DASD or copied from the SB buffer into the subpool and then written to the coupling facility.

Performance varies, depending upon:

- The cache option you choose.
- The number of databases utilized.
- The database block sizes.
- The number of sharing IMS/MVS images.

The "cache only changed data" option probably provides greater performance improvement than the "cache all data" option. If you specify the "cache only changed" data option, each data block from the selected database, once read from DASD, must be written to system memory. This action requires additional path length and cycles. The specific use of the database by your application program dictates whether this option is beneficial.

Data caching initialization processing is dependent upon the proper environment and the specification of the caching option in the subpool definition statement. The following actions are performed:

- Establish an element size of 2K for the coupling facility.
- Allocate a buffer prefix extension for caching subpools. The extension resides as a non-contiguous area and is anchored from the prefix. It consists of two parts:
 - A fixed length section for asynchronous processing.

- A variable length section for data transfer buffer lists. The number of buffer lists is dependent upon the subpool buffer size.

Example: A subpool buffer size of 6 KB will require three buffer lists and a subpool size of .5 KB or 1 KB will require one buffer list. A single buffer list will accommodate 2K of subpool buffer.

- Establish miscellaneous values for coupling facility data transfer parameters including:
 - Element and directory ratios
 - Buffer increment number (number of 256 byte segments)
 - Number of buffers in the buffer list.

The write data option is supported at the subpool level and permits the definition of each subpool with a write cache option.

Member DFSYDTx

The DFSYDTx member of IMS.PROCLIB specifies the OTMA descriptors that are built during IMS initialization. OTMA descriptors are optional. All parameters are delimited with a blank. The following is the format of a descriptor:

Column	Contents
1	Descriptor type. M for an OTMA descriptor.
2	Blank.
3-18	1- to 16-character client name, left-justified and padded with blanks if necessary. This is a required and positional parameter. Duplicate names are not allowed.
19	Blank.
20-27	Keyword: DRU= OTMA Destination Resolution exit routine name. This is a required parameter. Duplicate exit routine names are allowed.
73-80	Sequence numbers. These columns are ignored by IMS.

Up to 50 lines can be used in the specification of a descriptor. Columns 1 through 18 must be the same for each line.

Member DFS62DTx

LU 6.2 device descriptors are built during IMS initialization. They reside in member DFS62DTx in IMS.PROCLIB, where x is the IMS nucleus suffix. LU 6.2 device descriptors specify an LTERM that associates an output destination with an LU 6.2 device. The system administrator can additionally change applications that use alternate PCBs into applications that use LU 6.2 devices, without application coding changes.

No IMS generation or system definition changes are required to use an LU 6.2 device.

The record format is:

DFS62DTx

Position	Contents
1	Descriptor type ("U" for user descriptor) Although this column is not required for an LU 6.2 descriptor, it is included for consistency with ETO descriptors. It is not checked, and no error message is issued if it is omitted.
2	Blank
3-10	Contains the eight-character LTERM name, left-justified and padded with blanks, if necessary. This positional parameter is required, and is used as the descriptor name. A descriptor definition can be continued by repeating the descriptor name in subsequent records. Each individual descriptor can contain up to 50 records (excluding comment records).
11	Blank
12-72	Contains the parameter sets for the descriptor.
73-80	Ignored

DFS62DTx Descriptor Format

This is the format for the DFS62DTx descriptor. Do not code a parameter in the descriptor and then leave it blank such as (SIDE=b). If you do this, an error message is issued. Instead, omit it completely if you don't need to specify a DFS62DTx descriptor. A descriptor definition can be continued by repeating the descriptor name in subsequent records.

A ltermname parm1 parm2 parmn

where

U

is for user descriptor

ltermname

is the LTERM name

parm (1...n)

can be any of the following parameters: SIDE= MODE= TPNAME= SYNCLEVEL=
CONVTYPE=

Column	Column	Column
1	12	72
U DESC1	LUNAME=SOMELU	SIDE=SOMESIDE MODE=SOMEMODE CONVTYPE=SOMECONV
U DESC1	TPNAME=SOMETP	

Separate the parameters with a blank.

Keyword Descriptions

SIDE=

Is a one- to eight-character alphanumeric name identifying side information entry. (Side information refers to the system-defined values for CPI communications initialization.) If you specify an entry name on this keyword, the LUNAME, TPNAME, and MODE keywords are blank unless you also specify an overriding LUNAME, TPNAME, or MODE. If you do not specify an entry name, the default values apply to the LUNAME, TPNAME, and MODE keywords.

LUNAME=

Is a 1- to 17-alphanumeric character (uppercase) name of the destination LU

6.2 application program, which can be a 1- to 17-byte network-qualified LU name. An example of a network-qualified LU name is *netid1.luname1*. If SIDE= is specified, LUNAME overrides the LUNAME in the side information entry. The default is DFSLU.

Do not specify a network name for a dependent LU.

TPNAME=

Is a 1- to 64-character name of the transaction program to be scheduled. The default is DFSASYNC. If SIDE= is specified, TPNAME overrides the TPNAME in the side information entry.

Recommendation: Use upper-case characters for TPNAMEs so that the /ALLOCATE and /CHANGE commands can operate on the resulting messages and descriptors.

MODE=

Is a one- to eight-character name of the VTAM mode table entry to be used. If SIDE is specified, MODE overrides the mode name in the side information entry. The default is DFSMODE.

SYNCLEVEL=

Specifies whether the APPC/IMS sync level is confirmed (C) or not (N). The default value is C.

CONVTYPE=

Specifies whether the conversation type is basic (B) or mapped (M). The default value is M.

Specifying IMS Execution Parameters

You can specify control region execution parameters in the appropriate control region JCL, or in the IMS, DBC, and DCC PROCLIB members. Parameters specified in these members override any system generation. The parameters are not positional.

Database and PSB Exec Parameters for the Control Region

Three kinds of parameters in the IMS procedure that are related to database and PSB processing are:

- Pointers to database buffer requirements
- Overrides of predefined buffer sizes
- Performance options

Database Buffer Requirements

Use the VSPEC parameter to specify a two-character suffix for member DFSVSMxx in IMS.PROCLIB. This member predefines the buffer pool requirements for databases that use OSAM or VSAM as the access method. The IMS procedure generates a default value of 00 that points to a general- purpose buffer definition in member DFSVSM00, but you can change this suffix.

See “Specifying IMS Buffer Pools” on page 378 to specify subpool sizes.

Buffer Sizes

Several parameters enable you to redefine the size of buffer pools that hold DMBs and PSBs. Specify a value that represents the size as a number of 1024-byte blocks. For example, a value of 18 represents the size as 18432 bytes (18 x 1024 = 18432). IMS then rounds up this number based on the nearest whole number of

DB and PSB EXEC Parameters

pages on which these bytes fit. For example, if one page in MVS equals 4096 bytes, the value of 18 blocks would result in the allocation of 20480 bytes. This value overrides the value defined in system definition. The calculation is as follows:

Formula:

$$18 \times 24 = 18432/4096 = 4.5 \text{ pages (round up to 5)}$$
$$5 \times 4096 = 20480 \text{ bytes}$$

Performance studies often indicate a need to increase DMB or PSB buffer space. You can override the system definition values to implement this change.

If you use the DL/I address space option, two PSB pools exist: one in the MVS common area and one in the DL/I address space. Use the SASPSB parameter on the BUFPOOLS macro to specify sizes of the PSB pools during IMS system generation; use the CSAPSB and DLIPSB parameters on the IMS procedure to override the size of the PSSB pools. If the DL/I address space is not used, specify the size of the single PSB pool using the PSB= parameter on the BUFPOOLS macro; override this value with the PSB= parameter on the IMS procedure.

If Fast Path resources are used, an EPCB pool must contain extensions to Fast Path PCBs. For each PCB referencing a DEDB or MSDB database, an EPCB is required. This pool is used only by MPP regions, not by IFP or BMP regions.

Use the parameters of the IMS procedure in Table 26 to override the buffer pool sizes that were specified at system definition time with the BUFPOOLS macro.

Table 26. Overriding Buffer Pool Sizes with Database and PSB Parameters

IMS Procedure Parameter	System Definition Specification	
	Macro	Parameter
DMB	BUFPOOLS	DMB
DBWP	-	-
PSB	BUFPOOLS	PSB
PSBW	BUFPOOLS	PSBW
CSAPSB	BUFPOOLS	SASPSB(size1)
DLIPSB	BUFPOOLS	SASPSB(size2)
EPCB	BUFPOOLS	EPCB

The IMS procedure is generated with null value parameters for the above storage areas. The sizes default to their system definition values.

Performance Options

System definition lets you specify individual database DMBs and PSBs to be resident with the DATABASE and APPLCTN macros, respectively. You can override this specification with the RES parameter. The IMS procedure generates this as a null parameter; the default is RES=Y.

Request that all the control blocks are not to be made resident by specifying RES=N. You might do this when executing a test system or if storage is temporarily constrained.

Data Communication EXEC Parameters for the Control Region

Three kinds of parameters in the IMS procedure are related to data communication:

- Overrides of predefined buffer sizes
- Overrides and options particular to Message Format Service (MFS)
- Performance options

Buffer Sizes

By monitoring communication traffic you might decide to increase buffer space. Use the parameters of the IMS procedure in Table 27 to allocate virtual storage and override the buffer pool sizes predefined in system definition.

Table 27. Overriding Buffer Pool Sizes with Data Communication Parameters

IMS Procedure Parameter	System Definition Specification	
	Macro	Parameter
FBP ¹	BUFPOOLS	FORMAT(size1)
QBUF ²	MSGQUEUE	BUFFERS
RECA	COMM	RECANY

Note:

¹ Specify a value representing the number of 1024-byte blocks.

² By increasing the number of buffers (used for the message queue data sets), you can reduce the frequency of I/Os.

MFS Options

When you use MFS, you can adjust the capacity of the system by adjusting the availability of format blocks. Use the FRE parameter to specify the maximum number of active blocks in the message format pool. You need to coordinate this number with the value specified for the FBP pool size parameter.

Performance Options

The VAUT parameter offers a performance option to systems using VTAM. You can override the null value generated in the IMS procedure and specify the use of VTAM Authorized Path.

The NLXB parameter allows you to add parallel sessions during system startup. The number you specify is added to the number defined in the SESSION keyword of the MSPLINK macro.

The FESTIM parameter overrides the timeout value for front-end switching that is predefined in system definition. You specify the parameter on the COMM macro with a value from 1 to 300 seconds.

System Control and Performance EXEC Parameters for the Control Region

The positional parameters that affect control of the system resources follow.

Identification of Nucleus

The SUF parameter is generated as a null value for the IMS procedure, indicating that the default nucleus name has a one-character suffix of 0. If you are using an alternative control program, you must specify this parameter.

System Control and Performance EXEC Parameters

Active Regions

Use the PST parameter to override the expected number of regions that are to be in operation during the online execution. Additional regions can be dynamically allocated, up to the maximum allowable number permitted by your operating system. If you do not specify the PST parameter, the default number of regions is the number specified for the MAXREGN keyword in the IMSCTRL macro.

Performance Options

Three parameters contribute to general performance strategy. The FIX parameter specifies a two-character suffix for the member DFSFIXxx in IMS.PROCLIB. This suffix indicates which member should describe all modules and control blocks that are to be page fixed. The EXVR parameter allows you to page-fix buffers used for the management of message queues. The parameter is generated with a null value; you must specify EXVR=1 to indicate the page-fixing action. Similarly, the PRLD parameter specifies a two-character suffix for the member DFSMPLxx, where all preloaded modules are listed.

MVS Options

Several parameters apply only to the MVS operating system. The SRCH parameter allows you to take advantage of any special library structure to optimize the search for loaded modules. You can override the default value of 0 with a value of 1 if you want the JPA and LPA to be searched before IMS program libraries.

To reduce the amount of MVS CSA (common storage area) that IMS uses you can use one of the following techniques:

One technique is to specify LSO=Y on the IMS procedure to cause some control blocks and some IMS modules that are used for DL/I processing to be loaded into the control program's private storage. To allow for this transfer, use the following sum to increase the size of the control region:

Formula:

$$220K + \text{OSAM buffer pool} + \text{VSAM buffers} \\ + \text{enqueue/dequeue tables} + \text{system log buffers}$$

Another way to reduce IMS use of CSA is to take advantage of log buffer CSA VSCR (virtual storage constraint relief). Using CSA VSCR, several buffers can be moved out of CSA and placed above the 16-megabyte line into ECSA (extended CSA). The buffers that are relocated are the online log buffers, the batch log buffers, and the IMS Monitor buffers. This method not only frees CSA space, but it also allows these buffers more space in ECSA. To use this buffer relocation technique, specify LSO=S on the IMS procedure.

For details about specifying values on the LSO= parameter, see the description of the LSO= in "Chapter 4. Procedures" on page 189.

For operations, you can specify the output class of a spinoff dump using the SOD parameter. It is strongly recommended that you override the null value generated for this parameter for all MVS installations; it provides immediate analysis of the status of real storage after abnormal termination of the control region or of each dependent region.

DL/I Address Space

An additional variation of the local storage option is to use the DL/I separate address space. You do this by specifying LSO=S. This address space contains most of the DL/I code, control blocks, and database buffers for full-function databases. Again, MVS cross memory services are used.

Dump Formatting Options

In a DC environment, you can request the following types of dump outputs for errors that terminate IMS: SDUMP, SYSMDUMP, SYSABEND, or SYSUDUMP. To do this, specify the FMTO startup parameter in combination with MVS dump DD statements.

For SYSMDUMP, you should provide operational procedures for saving and formatting dumps; otherwise, you can overlay a SYSMDUMP if you must restart IMS before the previous SYSMDUMP is transferred.

You can also request dump outputs for some errors that do not terminate IMS. Your choice of dump depends on several factors: the type of failure, the FMTO parameter option, and the IMS spinoff and MVS dump DD statements that have been selected. For more information about the FMTO parameter and about using dumps in these situations, see “Chapter 5. IMS Dumping and Dump Formatting Options” on page 287.

IRLM Options

If an execution of the IMS online system is to use IRLM as lock manager or participate in block-level sharing, specify IRLM=YES. Specify the MVS subsystem name for the IRLM associated with this control region on the IRLMNM parameter.

Related Reading: For details of IRLM requirements, see *IMS/ESA Administration Guide: System*.

If your online system includes Fast Path, use the UHASH parameter with the IRLM to specify the name of an alternative hashing module. You must specify the appropriate name if you did not use the default name of DBFLHSH0 in system definition. You cannot change this name across a restart. If multiple IMS subsystems are participating in data sharing and are using this option, the module must be identical in all systems.

Subsystem Identification Parameters

Use the SSM parameter to reference a member in IMS.PROCLIB which identifies the DB2 subsystems that can be accessed from application programs executing in dependent regions. The one- to four-character suffix that you specify, together with the currently assigned name for IMSID, forms the member name. The member contains entries, each identifying a DB2 subsystem (its MVS subsystem name). All DB2 subsystems to be accessed from programs executing in dependent regions must have corresponding entries.

Depending on its entry, a region can access all, some, or none of these subsystems. To allow a dependent region to access all subsystems identified to the IMS control region, do not specify the SSM parameter for the dependent region or have the SSM entry specify the same member as the IMS procedure. To allow a dependent region to access only selected DB2 subsystems, specify an SSM parameter on the dependent region procedure that points to a member containing only those specific subsystems. To prevent a dependent region from having access to any subsystem identified to the IMS control region, specify a member containing no entries.

See “Chapter 10. Accessing DB2 Databases with IMS” on page 419 for details about how to specify the individual entries and construct the IMS.PROCLIB member.

Recovery-Related EXEC Parameters

Recovery-Related EXEC Parameters for the Control Region

The WADS parameter indicates whether the IMS online system is to use a single write-ahead data set (WADS) or dual data sets. The WADS helps protect the integrity of the online logging. In case of write errors, additional data sets can be defined up to a total of ten. Specify WADS=S for single recording or WADS=D for dual recording.

Use the ARC parameter to specify whether automatic archiving of the online log data sets (OLDS) is to be performed. Automatic archiving is recommended; however, you can arrange for the MTO to monitor the availability of the OLDS and perform archiving when necessary. Specify ARC=0 for no automatic archiving; otherwise, specify a one- or two-digit number representing the number of full (closed) OLDSs that invoke automatic archiving for those data sets.

Specify a name (do not accept the default) for the start procedure for the DBRC region on the DBRCNM parameter. The default is a null parameter indicating a procedure name of DBRC. Also as part of your installation's recovery strategy, the AUTO parameter specifies whether to use automatic restart of the IMS online system. You need to override the generated default value (N) by specifying AUTO=Y if you plan to use automatic restart.

Two parameters, QTU and QTL, are associated with protection of the message queues. Use them to adjust the values in the message queue space notification exit routine (DFSQSPC0). Given the number of records reserved on each message queue data set for automatic shutdown (specified in system definition with the SHUTDWN parameter on the MSGQUEUE macro), a finite number of residual records are available for message use. The notification exit routine keeps track of the current level of utilization. If the usage exceeds the upper limit (expressed as a percentage of records used), the MTO is notified so that steps can be taken to reduce the backlog. The MTO is also notified if the usage falls below the lower limit.

The QTU and QTL values are one or two-digit percentages. QTU overrides the upper threshold value and QTL overrides the lower one (100 is allowed for QTU). Both upper and lower threshold values are generated with null values; the defaults are those of the IBM-supplied exit routine—75% and 60%, respectively. Unless your installation establishes its own exit routine to control the MTO notification, you should probably use these defaults. In the long term, you can tune them to fixed values based on the feedback from the statistics given in the /DISPLAY POOL command; enter this command during peak periods of activity.

Related Reading: Details of the IBM-supplied notification exit routines are in *IMS/ESA Customization Guide*.

Security-Related EXEC Parameters for the Control Region

Several parameters control the kind of security checking that is done during the current execution by acting as switches for different types of security; moreover, they determine how much flexibility the MTO has to override the choice of security checking. You must coordinate the setting of these switches with both overall security design and operational procedures. The parameters are TRN, SGN, RCF, and ISIS.

All the values generated for the IMS procedure parameters specify **no** security. You need to reset them to match the security design. You must also coordinate the level of the security tables with the suffix identifier for the nucleus.

Security-Related EXEC Parameters

Related Reading: Operational restrictions for the MTO, and the parameter values and their meaning are discussed in *IMS/ESA Administration Guide: System*.

The IMSID parameter is partially related to security and to operations. The one- to four-character value that you specify uniquely identifies the control region. Dependent regions executing under control of this nucleus must specify the same identifier. The generation default is IMSA.

Any message sent to the console from subsystem execution is identified by this name; therefore, the name chosen should be unique to other subsystems executing in the MVS system, including any batch executions. The IMSID name should not be the same as the procedure name in order to avoid confusion as to the source of the message.

IMS Message Processing Region Parameters

This section does not apply to DBCTL.

The PARM= positional parameters for the message processing region's EXEC statement are shown in Table 28. See the DFSMPR procedure for more information.

Table 28. Categories and Purpose of Message Processing Region Parameters

Category	Parameter	Purpose
Database and PSB	PCB	Specify size of pool for PSB copy
	VALCK	Check validity of addresses in DL/I calls
	NBA	Specify Fast Path database buffers
	OBA	Specify Fast Path overflow buffers
Data Communication	CL1	Specify priority of message class
	CL2-4	Specify up to three lower-priority message classes
Region Control and Performance	LOCKMAX	Turn off locking limitations
	MSG	Specify MSG for message region
	OPT	Coordinate region start with control region
	OVLA	Retain overlay supervisor
	PRLD	Specify suffix of DFSMPLxx for program preload
	PREINIT	Specify suffix of DFSINTxx for modules to be preinitialized in readiness for takeover
	STIMER	Specify timer options
	DBLDL	Specify maximum number of IMS.PGMLIB directory entries to be saved
	SOD	Specify output class for spinoff dump
	VSFY	Specify suffix of DFSVFLxx for programs used with virtual fetch
VFREE	Free storage for programs used with virtual fetch	
SSM	Specify access to DB2 subsystems	

IMS Message Processing Region Parameters

Table 28. Categories and Purpose of Message Processing Region Parameters (continued)

Category	Parameter	Purpose
	ALTID	Specify alternate IMS system
	PWFI	Activate pseudo wait-for-input
	APARM	Specify application parameter
Recovery and restart	SPIE	Allow user program SPIE during DL/I call
	TLIM	Specify number of abnormal terminations allowed
Security options	AGN	Specify application group name
	IMSID	Override IMS subsystem identifier

PSB-Related EXEC Parameters for a Message Processing Region

The PCB parameter is an optional parameter that specifies the size of the interregion communication area. This area holds the PCBs included in the PSBs that are used by programs executing in this region. If you do not specify this parameter, the default value is 000. This allocates an area large enough to hold all the PCBs for the largest PSB defined to this IMS system and present in the active IMS.ACBLIBA/B. To specify a value, enter the number of 1024-byte blocks as a three-digit number (for example, 001).

The default value for the VALCK parameter signifies that address validity checking is not to be performed for DL/I calls issued by the application programs in this region. (An address is invalid if it is either lower than the lowest address not in the MVS nucleus or higher than the highest address in virtual storage.) With adequate testing of and controls over the DL/I call parameter coding, validity checking should not be necessary.

Data Communication EXEC Parameters for a Message Region

A message region requires the set of four parameters, CL1, CL2, CL3, CL4, to specify transaction classes. Express each class as a three-digit number. The first message class for the region causes all messages assigned to that class to be selected first as eligible for scheduling. Only when all possibility of scheduling a transaction in that class has been exhausted does scheduling begin for the second message class. Priorities determine the order that programs are to be selected for scheduling into the region. The message classes you specify need to be coordinated with your transaction scheduling rules and the numbers entered with the PRTY keyword on the TRANSACT macro.

Region Control EXEC Parameters for a Message Region

MSG is the first positional parameter for a message processing region. When transactions are encountered on the queue, programs are automatically scheduled in these regions, if the message class priority is suitable.

The parameter OPT helps you control the region startup. If, for some reason, the control region is not active or is terminating when the MPP region is invoked, you can have the MTO decide whether to start the MPP region again (the default), let it wait until the control region is ready, or cancel it.

Use STIMER to invoke timer facilities. If you use the non-default LSO option, LSO=S, both application program and DL/I processing times are recorded. Most DL/I processing time is not included when you use the default, LSO=Y. A value of 1 causes a timer sequence to be issued once for each DL/I call that can adversely

IMS Message Processing Region Parameters

affect the performance of the region. This parameter is used in conjunction with the TIME parameter on the //JOB statement. Because a timeout of a message region causes an abnormal termination of the message region, it is recommended to have processing limits coded within the application program.

Use the SSM parameter to point to a member in IMS.PROCLIB that identifies the DB2 subsystems that can be accessed from this MPP region. To prevent any access to a DB2 subsystem from an MPP scheduled into this region, use the name of a null member (no entries). The default, a null value, allows the MPP region to attach to any of the subsystems declared to the IMS control region. If necessary, coordinate this parameter with the corresponding SSM parameter in the IMS procedure. Refer to the description in “System Control and Performance EXEC Parameters for the Control Region” on page 367.

Use the APARAM parameter to specify execution time parameters unique to this dependent region. This parameter specifies a character string for the application program or Data Capture exit routine. The APARAM can be used to specify the frequency for checkpoint calls. The APARAM can also be used to pass parameters to the Data Capture exit routine to indicate whether the data should be captured.

The maximum length of the parameter is 32 characters; the parameter must be enclosed in single quotes (') if special characters are used; embedded commas are not allowed. The INQY call with the ENVIRON subfunction is used to receive the APARAM character string.

Related Reading: For more information, see *IMS/ESA Application Programming: Database Manager*.

Use the SOD parameter to specify the output class for a spinoff dump. In this way, you can obtain a printed copy of the storage immediately at abnormal termination of the message region immediately rather than being placed on general SYSOUT queues.

Performance-Related EXEC Parameters for a Message Region

Four performance-related options are available. The parameter OVLA allows you to retain a copy of the overlay supervisor in the message region. The default is to have the overlay control loaded each time you schedule an overlay application program in the region. If you have arranged the message priorities so that an overlay program is likely to be scheduled in the region, you should use this option.

Use the DBLDL parameter to minimize the cost of program loading by increasing the number of program directory entries maintained in the message region. The effect is to reduce the I/O to the program library (and the search) to obtain the direct address of the program. The default is 20.

Use the PRLD parameter to specify the two-character suffix of the member DFSMPLxx in IMS.PROCLIB. This member contains the list of programs that are to be preloaded. The effect is to reduce the repetition of I/O for program load. The region address space retains the modules.

PREINIT allows you to specify the two-character suffix of the member DFSINTxx in IMS.PROCLIB. This member contains the preinitialization modules to receive control.

Two parameters apply only to the MVS operating system using virtual fetch to manage program loading. To invoke virtual fetch, specify the VSFY parameter. This

IMS Message Processing Region Parameters

parameter specifies the two-character suffix of the member DFSVFLxx in IMS.PROCLIB, which lists the modules to be managed by virtual fetch.

The type of program load selected depends on the order of the following:

1. Preload list
2. Virtual fetch list
3. Standard MVS fetch

If you specify a module for virtual fetch that was previously in the preload list, you need to delete that module name from the DFSMPLxx member. You can rebuild virtual fetch's VIO data set and hash table to reflect changes made to the load modules in your source libraries. The refresh process makes all updates available to virtual fetch users at the same time.

The VFREE parameter instructs IMS to release or retain virtual storage at program termination. If you use the program infrequently, and you need dependent region virtual storage, it is recommended that you specify a value Y for the VFREE parameter to free the storage. If you are optimizing for programs that are not used continuously by the message processing region, the combination of freeing virtual storage and virtual fetch I/O is appropriate.

The fourth performance-related parameter is the PWFI= (Pseudo WFI) parameter. Specifying PWFI=Y can potentially reduce CPU time by eliminating the termination and rescheduling of resources. On a message GU call if MODE=SNGL is specified on the TRANSACT macro and no more messages are available, IMS checks to see if other work needs to be done for the region. If no other work is available and the resources owned by the region are not needed by another region with work to do, the region becomes idle and waits until another message is queued (wait-for-input mode). It does not return a "QC" status code to the application program. If the new message is for the transaction that is scheduled in the region, IMS returns the new message to the application program without having to terminate and reschedule the resources.

If the next message that comes in is not for the transaction that is scheduled in the region, but it can be processed by the region, there is a delay before the new transaction can be processed. Because termination of the previous message is delayed until after the new message comes in, the region must go through termination before it can schedule the new transaction.

Recovery-Related EXEC Parameters for a Message Region

The TLIM parameter addresses a problem with an application program that causes an abnormal termination. Because the program can be scheduled many times into a region as a result of transactions in the queue, you need to be able to stop the operation of this region. The value for TLIM specifies the limiting number of abnormal terminations permitted. In the case where the application program has a SPIE in effect, the SPIE option allows it to be left on during the DL/I call or for it to be turned off during the DL/I call and reinstated when returning to the application program. For performance reasons, it is not recommended to turn it on and off for each DL/I call. With PL/I Release 5 or later, you can use the PL/I SPIE facility without having IMS reset the SPIE on each DL/I call.

Security-Related EXEC Parameters for a Message Region

The AGN parameter allows you to specify the application group name, which is associated with a set of transactions, PSBs, or LTERMs that are authorized to be used by this region. Declare these resources and the AGN in the Security Maintenance utility input to authorize them to be used. Include the same name to

IMS Message Processing Region Parameters

invoke IMS resource access security for this region. The DFSMPR procedure generates this as a null parameter. If other regions are being controlled through the use of AGNs, you risk permitting unauthorized access by programs executing in this region if you omit this parameter.

The parameter IMSID is related to security and to operations. You specify the 1- to four-character identifier for the name of the control region (the name given for the IMSID parameter). If the value does not match the current IMSID of any operating control region, the message region is not scheduled.

IMS Batch Message Processing Region Parameters

The PARM= positional parameters for the BMP region's EXEC statement are shown in Table 29. See the IMSBATCH procedure for more information.

Table 29. Categories and Purpose of BMP Region Parameters

Category	Parameter	Purpose
Database and PSB	MBR	Specify name of batch message program
	PSB	Specify PSB name
	TEST	Check validity of addresses in DL/I calls
	NBA	Specify Fast Path database buffers
	OBA	Specify Fast Path overflow buffers
Data Communication	IN	Specify an input transaction queue
	OUT	Specify an output transaction queue
Region Control and Performance	BMP	Specify BMP for batch message region
	OPT	Coordinate region start with control region
	LOCKMAX	Specify locking limitations
	PRLD	Specify suffix of DFSMPLxx for program preload
	PREINIT	Specify suffix of DFSINTxx for modules to be preinitialized in readiness for takeover
	STIMER	Specify timer options
	CPUTIME	Specify processing time limit
	PARDLI	Specify region for DL/I processing
	DIRCA	Specify interregion communication size
	SSM	Specify access to DB2 subsystems
Recovery and Restart	ALTID	Specify alternate IMS system
	APARM	Specify application parameter
	SPIE	Allow user program SPIE during DL/I call
	CKPTID	Specify checkpoint for program restart
	Security options	AGN
IMSID		Override IMS subsystem identifier

PSB-Related EXEC Parameters for a Batch Message Region

The MBR parameter specifies the name of the program and is often the same as the PSB name. This parameter is required. The BMP has flexibility in using a

IMS Batch Message Processing Region Parameters

program and PSB combination. This allows you to test modifications of the BMP using a temporary program name. You can also use a different PSB with the same program.

The PSB parameter is optional if it matches the MBR name. If an APPLCTN macro has been included for the PSB specifying BATCH as the program type, the PSB parameter specifies one of several PSBs for a given BMP. You should identify application programs that have the ability to process several types of transactions. These programs often require larger amounts of virtual storage, and you can need to adjust the size of the region.

The TEST parameter is required, but the IMSBATCH procedure generates a 0 to specify that validity checking of the addresses in a DL/I call is **not** performed. (An address is invalid if it is either lower than the lowest address not in the MVS nucleus or higher than the highest address in virtual storage.) Adequate testing of the program and controls over the DL/I call parameter list coding should make the generated option (no validity checking) appropriate.

The NBA parameter specifies the number of Fast Path database buffers to be made available. These buffers are taken from the global pool defined by the DBBF= parameter in the IMS control region procedure.

The OBA parameter specifies the number of additional database buffers to be made available in the CSA/ECSA. Fast Path allocates only the largest OBA specification from all currently active regions.

Data Communication EXEC Parameters for a Batch Message Region

For the batch message program, you have the flexibility of declaring that the input transaction queue is to be made available to the program at execution time. You do this by specifying one transaction code as the value for the IN parameter.

Some BMP programs do not access a message queue; however, they have a requirement to send output to a terminal or to generate transactions to be processed by other application programs. Specify the LTERM name or transaction code (as appropriate) on the OUT parameter.

A null value for the IN parameter prevents the program from accessing message queues. When you specify the transaction code for the IN parameter, the program has no restrictions on generated transactions or output messages.

Region Control EXEC Parameters for a Batch Message Region

BMP is the first positional parameter for a batch message processing region. These regions are not scheduled automatically. They must be invoked by the operator.

The OPT parameter helps you control the start of a batch message region. If the control region is not active when the BMP region is invoked, you must decide whether to wait for a control program, cancel the batch message region, or ask the operator to make the decision. This might occur when JCL in the MVS job stream for the BMP region is scheduled before the control region has completed its initialization or when the control region is terminating. Specifying 'wait' is a risk because the MVS resource is reserved until the control region resumes. If you are starting the region from the master terminal, the default generated for the IMSBATCH procedure is satisfactory.

IMS Batch Message Processing Region Parameters

The optional PRLD parameter specifies the two-character suffix for the member DPSMPLxx in IMS.PROCLIB, which lists the preloaded modules. It is optional; it is generated as a null parameter for the IMSBATCH procedure. For BMP regions that do not have other programs scheduled in them, PRLD provides no performance gain.

The PREINIT parameter specifies the two-character suffix of the member DFSINTxx in IMS.PROCLIB. This member contains the preinitialization modules that are to receive control.

The optional STIMER and CPUTIME parameters are generated as null parameters, their defaults are no use of the timer.

The optional PCB parameter specifies the size of the interregion communication area. This area holds the PCBs included in the PSBs that are used by programs executing in this region. If you do not specify this parameter, the default value is 000. This allocates an area large enough to hold all the PCBs for the largest PSB defined to this IMS system and present in the active IMS.ACBLIBA/B. To specify a value, enter the number of 1024-byte blocks as a three-digit number (for example, 001).

Use the SSM parameter to reference a member in IMS.PROCLIB that identifies the DB2 subsystems that can be accessed from this BMP region. To prevent any access to a DB2 subsystem from the BMP, use the name of a null member (no entries). The default (a null value) allows the BMP region to attach to any of the subsystems declared to the IMS control region. If necessary, coordinate this parameter with the corresponding SSM parameter in the IMS procedure. Refer to the description in "System Control and Performance EXEC Parameters for the Control Region" on page 367.

The FMTO parameter and the type of MVS dump DD statements selected determine whether IMS dumps are formatted online or offline.

Use the APARM parameter to specify execution-time parameters unique to this dependent region. This parameter specifies a character string for the application program or Data Capture exit routine. For batch or BMP regions, the APARM can be used to specify the frequency for checkpoint calls. The APARM can also be used to pass parameters to the Data Capture exit routine to indicate whether the data should be captured.

The maximum length of the parameter is 32 characters; the parameter must be enclosed in single quotes (') if special characters are used; embedded commas are not allowed. The INQY call with the ENVIRON subfunction is used to receive the APARM character string.

Related Reading: For more information, see *IMS/ESA Application Programming: Database Manager*.

Recovery-Related EXEC Parameters for a Batch Message Region

For a BMP program, you can use the CKPTID parameter to set a restart position for the program processing. The IMSBATCH procedure generates CKPTID as a null parameter. To invoke restart, create a special version of the procedure containing the exact checkpoint identification. This can be a one- to eight-character extended checkpoint identifier generated by the program itself, or a twelve-character time-stamp identifier issued within a message from checkpoint processing. Alternatively, you can specify the value *LAST* as the identifier to cause the last

IMS Batch Message Processing Region Parameters

recorded checkpoint to be used. One technique is to restart BMPs with JCL entered in the system reader and have the CKPTID symbolic parameter coded for the EXEC statement.

Related Reading: For more details regarding BMP restart for BMPs that use extended checkpoint, refer to *IMS/ESA Operations Guide*.

In the case where the application program has a SPIE in effect, you specify whether the SPIE is to remain on or off during the DL/I call. Negated SPIEs are reinstated before returning to the application program. For performance reasons, it is recommended that the SPIE not be turned on and off for each DL/I call. With PL/I Release 5 and later releases, you can use the PL/I SPIE facility without having IMS reset the SPIE on each DL/I call.

Security-Related EXEC Parameters for a Batch Message Region

The AGN parameter allows you to specify the application group name, which is associated with a set of transactions, PSBs, or LTERMs that are authorized to be used by this region. Declare these resources and the AGN in the Security Maintenance utility input to authorize them to be used. Include the same name to invoke IMS resource access security for this region. The IMSBATCH procedure generates the AGN as a null parameter. If other regions are being controlled through the use of AGNs, you risk permitting unauthorized access by this batch message program if you omit this parameter.

The parameter IMSID is related to security and to operations. You specify the one-to four-character identifier for the name of the operational IMS DB/DC environment. If the value does not match the current IMSID of any operating control region, the batch message region is not scheduled.

Specifying IMS Buffer Pools

Three types of buffer pools control the IMS DL/I buffering services: the VSAM shared resource pool, the OSAM buffer pool, and the OSAM Sequential Buffer (SB) pool. This section describes the parameters that define the size and content of these pools. Each data set can have no more than one open ACB (VSAM access method control block).

VSAM constructs the VSAM shared resource pool based on parameters provided by the VSAM BLDVRP macro, which is issued during IMS initialization. This pool contains buffers to be used for VSAM data sets (both index and data components) and the input/output-related control blocks necessary to perform VSAM requests. The buffers are combined in subpools. All buffers within a subpool are of equal length.

The OSAM buffer pool is required for IMS online and batch operations. This pool contains buffers to be used for OSAM data set data components and the input/output-related control blocks necessary to perform OSAM requests. The buffers are combined in subpools. All buffers within a subpool are of equal length.

IMS dynamically constructs the OSAM Sequential Buffer (SB) pool when application programs or utilities using SB are active. To allow use of SB by online applications, you must provide a SBONLINE control statement in the DFSVSMxx member of the IMS.PROCLIB data set for an IMS DB/DC or DBCTL environment.

Specifying Buffer Pools

Each application program or utility using SB has its own SB buffer pools. You can control the use of SB by an application program or utility with PSBGEN, with control statements in the //DFSCCTL data set, and with an SB Initialization exit routine.

Related Reading: See the explanation on “How to Request the Use of SB” in *IMS/ESA Administration Guide: Database Manager*.

VSAM Subpool Definition

Define VSAM subpools using the VSRBF statement, which is described in “Defining VSAM Subpools” on page 342. Specify the required subpools in the DFSVSAMP data set for batch environments or in the DFSVSMxx member of the IMS.PROCLIB data set for DB/DC environments. The VSAM buffers and control blocks are above the 16-megabyte line in most environments. For utilities that run in a ULU or UDR region in a batch environment, the VSAM buffers are below the 16 MB line.

The minimum number of subpools is one and the maximum is eleven. The minimum IMS-calculated number of buffers in a subpool is three, and the maximum is 32767. This calculation is based on region type, number of PCBs, number of data set groups, and database organization.

A database is in sequential mode if **all** of the following conditions are true:

- The database operates in a batch type region (DLI or DBB)
- The database is referred to by only 1 PCB in the PSB
- The database organization is HISAM, single-segment HISAM, INDEX, or HIDAM accessed through unqualified GN calls

Buffer sizes can be .5K, 1K, 2K, 4K, 8K, and multiples of 4K up to 32K. IMS accepts any valid CI size up to 30K, but always uses a buffer size equal to or larger than the CI size. For example, a 30-K CI uses the 32-K subpool. If no VSAM files are used, the VSAM subpools need not be defined. A buffer handler pool is always built.

During DL/I database open, a data set is assigned a specific buffer subpool based on the CI size. The CI size must be equal to or less than the buffer size for the subpool assigned. The data and index components of a key-sequenced data set (KSDS) can be assigned to different subpools if their CI sizes are different and corresponding subpools exist. VSAM can assign a larger CI size to the INDEX component than to the DATA component when defining KSDS data sets that have small record sizes. For example, VSAM can assign a 512-byte DATA CI size and a 4096-byte INDEX CI size. You must be sure to define subpools that accommodate both INDEX and DATA components of a KSDS. (A listing of the VSAM catalog shows the CI sizes assigned to the components.) A single subpool can be defined with buffers large enough to contain the longest CI, or you can define several subpools that more nearly fit the different-sized CIs used by the programs.

Related Reading: Refer to *IMS/ESA Administration Guide: System* for additional information on buffer pool structure and buffering techniques.

OSAM Subpool Definition

OSAM subpools are defined using the IOBF and DBD statements as described in “Defining OSAM Subpools” on page 347. Specify the required subpools in the DFSVSAMP data set for batch environments or in the DFSVSMxx member of the IMS.PROCLIB data set for DB/DC environments. You can define multiple OSAM subpools as having the same buffer size (that is, the length parameters specified on

OSAM Subpool Definition

the IOBF subpool definition statements are identical); then, you can direct given data sets to specific subpools. To take advantage of this flexibility, the desired subpool must have a user-defined ID (via the ID subparameter) and a DBD statement coded to identify the data set.

If you do not want to assign data sets to specific subpools, do not code the DBD subpool definition statement. If you define multiple OSAM subpools having the same buffer size and ID, IMS generates only one subpool for that buffer size. Within this subpool, the number of buffers is equal to the sum of the number of buffers specified for these subpools. The OSAM buffers and control blocks are above the 16-megabyte line. In a batch environment, the OSAM buffers and control blocks are located in the private address space. You must allocate a region size that is big enough to accommodate your OSAM buffer requirements. For utilities that run in a ULU or UDR region in a batch environment, the OSAM buffers are below the 16-megabyte line.

OSAM Buffer Pool Compatibility Definition

The OSAM buffer pool consists of one or more subpools defined through IOBF parameter statements. These statements are required in a DB/DC environment. Default member DFSVSM00 is used if the VSPEC parameter is not coded, but these statements are optional in a batch environment.

Omitting IOBF statements in a batch environment causes the OSAM subpools to be generated based on the following factors:

- The DMBs loaded for execution
- The values specified by the BUF parameter on the EXEC statement

The BUF parameter specifies the size of the OSAM buffer pool that is to be used in the subpool compatibility definition calculations. The size specified by the BUF parameter does not necessarily define the actual size of the buffer pool. The specified size influences the eventual buffer pool size.

The buffer pool is generated as follows:

- IMS inspects each DMB loaded for execution for an OSAM DCB. The buffer length from the DCB is rounded up to the next appropriate subpool size (.5K, 1K, 2K, and so forth) and is used to create a potential subpool. DCBs with similar length requirements increment the occurrence count for that subpool.
- The potential number of buffers for each potential subpool is derived from the ratio of actual to least occurrences, multiplied by 4. The minimum number of buffers allowed per subpool is four.
- IMS allocates the actual number of subpool and buffers according to the previous ratio allocation using the buffer pool size specified by the BUF parameter. The objective is to allocate at least four buffers per subpool without exceeding the specified size. The process begins with the subpool with the largest buffer size and continues to the subpool with the smallest. Depending on the specified buffer pool size, it might not be possible to generate all required subpools. IMS allocates only those subpools (largest to smallest), with a minimum of four buffers each, that fit in the specified size. A minimum of one subpool with a minimum of four buffers is allocated regardless of the buffer pool size defined. This subpool has a buffer size capable of handling the largest buffer size requirement.

Example: The BUF parameter is specified as 10 KB. A scan of the loaded DMBs and the block size required by the OSAM DCBs indicates two subpools are

OSAM Buffer Pool Compatibility

required: one with 1-K buffers, and one with 4-K buffers. If each subpool has the minimum number of buffers, the total storage for the buffer pool is 20K. The size specified (by BUF=) is 10000 bytes. To support all requests, the subpool having 4-K buffers must be allocated. The subpool with 1-K buffers is not allocated. The buffer pool is allocated having one subpool with four 4-K buffers. The size is 16K (plus control areas), even though 10000 bytes are specified.

Specifying VSAM and OSAM Subpools

The following example shows the control statements necessary to construct a single OSAM subpool with four 2K-buffers each and two VSAM subpools. The first subpool contains four 2048-byte buffers, and the second subpool contains fifteen 1024-byte buffers. The new KSDS records use mass insert.

```
IOBF=(2K,4,N,N)
VSRBF=2048,4
VSRBF=1024,15
OPTIONS,INSERT=SEQ
```

For information on specifying the above control statements, see “Member DFSVSMxx” on page 339.

For IMS batch environments, the //DFSVSAMP data set provides the control statements listed above. Additionally, the //DFSVSAMP data set provides the GTF trace option. GTF trace records reflect the stream of I/O requests to the OSAM buffer handler.

Related Reading : For more information on GTF trace records, refer to *IMS/ESA Administration Guide: System*. In an IMS environment, the DFSVSMxx member of the IMS.PROCLIB of the IMS system provides the statements. You should not place DFSVSAMP in the same PDS as your user application files. This causes an ABEND0C4 in DFSRTM00.

Specifications for OSAM Sequential Buffering

The SBONLINE control statement allows use of OSAM Sequential Buffering (SB) in an IMS DB/DC or DBCTL environment. For information on specifying the SBONLINE control statement, see “Specifying Sequential Buffering for an Online System” on page 350.

One or more of the following factors controls the use of SB by a particular application program or utility:

- PSBGEN
- Control statements in the //DFSCTL data set
- An SB Initialization exit routine

Related Reading: For more information, see *IMS/ESA Administration Guide: Database Manager*.

See also “Specifying Sequential Buffering Control Statements” on page 383 for more information on how to request use of SB through control statements in the //DFSCTL data set of the IMS batch region or IMS dependent online region.

Making IMS and IMSRDR Procedures Accessible to MVS

The IMS and IMSRDR procedures initiate the IMS control program region and message regions. These procedures are placed in IMS.PROCLIB by system definition and must be added to SYS1.PROCLIB.

MVS with JES2 or JES3: Concatenate IMS.PROCLIB with SYS1.PROCLIB, or move the following procedures to SYS1.PROCLIB:

- IMS
- IMSRDR
- DBC
- DBRC address space procedure (as specified in DBRCNM= in the IMSCTRL macro)
- DL/I address space procedure (as specified in DLINM= in the IMSCTRL macro)
- DXRJPROC - IRLM procedure
- DFSMPR
- DFSWTnnn

The data set IMS.PROCLIB should be in the master catalog, otherwise, ensure that the volume serial number and unit are on the DD statement. If this is not done, JES2 is unable to initialize.

These procedures are described in “Chapter 4. Procedures” on page 189.

Any dependent region jobs (IMSMSG, IMSWTnnn) to be specified in the IMS /START REGION command must be added to IMS.JOBS.

Organizing PL/I Modules for Use with the PL/I Optimizer

You can decrease response time for those IMS applications that use the PL/I Optimizer by organizing the PL/I modules. Some organizational recommendations follow:

- Use several different program libraries, one for each region, putting only those modules required by the application in the library. Include in that library all supporting modules (such as the PL/I transient library modules).
- Concatenate the PL/I library into the message region STEPLIB.
- Put the required supporting modules in the link pack area. This is the recommended long-term solution for a virtual environment.
- Use IMS PRELOAD on the above modules.

Attention: Do not use the PL/I Optimizing Compiler for multitasking during link-editing. Do not use SYS1.PLITASK as a SYSLIB data set.

Coding COBOL Programs That Run under IMS

Related Reading: For information on recommended compiler options, link-edit attributes, and run-time options, see “Coding COBOL Programs that Run under IMS” in *VS Cobol II Application Programming Guide*. The LIBKEEP=YES run-time option is recommended.

See also *Programming Language Considerations in an Information Management System (IMS) Environment*.

Specifying Sequential Buffering Control Statements

This section describes the syntax and keywords of the six sequential buffering (SB) control statements: SBPARM, SBIC, SBCO, SBSNAP, SBESNAP, and SNAPDEST.

Related Reading: For a more general description of SB concepts and terminology, refer to “Program Specification Block (PSB) Generation” in *IMS/ESA Utilities Reference: System* and *IMS/ESA Administration Guide: Database Manager*.

Sequential buffering can be invoked by default for IMS utilities, such as online image copy, without requiring SB control statements, through the use of the SB Initialization exit routine.

Related Reading: For the description of the routine refer to “Sequential Buffering Initialization Routine” in *IMS/ESA Customization Guide*.

SB control statements allow you to:

- Specify which I/O operations should be buffered by the SB buffer handler
- Override the default number of buffer sets to increase buffering performance
- Experiment with SB specifications without changing exit routines or regenerating the PSB
- Use SB problem determination aids

SB control statements are in a data set with the ddname of //DFSCCTL. You can provide a //DFSCCTL DD statement in the JCL of the IMS batch or IMS dependent online region. The //DFSCCTL DD statement must point to a sequential data set or a member of a partitioned data set. The record format of the //DFSCCTL data set must be F, FB, or FBS.

Syntax of SB Control Statements

SB control statements are contained in 80-character records. Positions 72-80 are ignored.

Each SB control statement must begin on a new record. The first record must contain at least the statement name. IMS considers the first non-blank character the start of the statement. The first word on the statement must be the statement name, such as SBPARM and SBIC, followed by one or more blanks.

Depending upon the type of SB control statement, one or more keyword parameters follow the statement name. Keywords must be followed immediately by an equal (=) sign and a value. If you specify multiple keywords, separate them with commas. If you specify multiple values, enclose them in parentheses. Individual keyword parameters must be completely contained on a single record.

A blank or a /* signals the end of a statement. You can continue SB control statements on multiple records. Use a comma at the end of a line to indicate continuation.

An asterisk (*) in the first position on a record indicates a comment. Elsewhere on a record, comments begin with /*.

Two examples of proper syntax follow:

Syntax of SB Control Statements

```
SBPARM ACTIV=COND,DB=SKILLDB,BUFSETS=6 /*Comment

SBPARM ACTIV=COND, /*This is a comment
          DB=SKILLDB, /*This is a comment
          BUFSETS=6 /*This is a comment
```

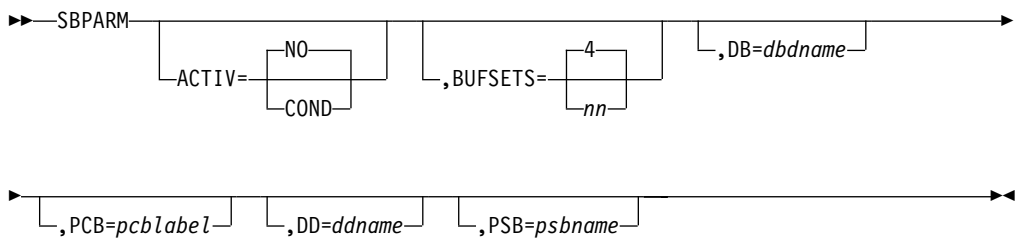
SB Parameters (SBPARM) Control Statement

The SBPARM control statement is used to:

- Specify which I/O operations require sequential buffering (I/O operations can be specified by database, PCB, ddname, and PSB)
- Override SB default parameters

All SBPARM parameters are optional.

The format of the SBPARM control statement is shown below.



Default values are underlined. The default value, however, might have been overridden by other specifications. If you provide the same SB option or parameter in more than one place, the following priority list applies (item 1 has the highest priority):

1. SB control statement specifications (the n th control statement overrides the m th control statement, where $n > m$)
2. PSB specifications
3. Defaults changed by the SB Initialization exit routine

If you specify multiple SBPARM control statements applying to the same I/O operation, the n th control statement overrides the m th one (where $n > m$). This lets you specify defaults on the first statement which can be overridden by subsequent statements.

The SBPARM control statement has two types of keyword parameters:

1. Option parameters, such as ACTIV and BUFSETS, which set and override an SB option or parameter, respectively
2. Qualification parameters, such as DB, PCB, DD, and PSB, which specify the I/O operations to which the SBPARM statement applies

Option Parameters

Option parameters set or override an SB option or parameter. If an option parameter is not specified, the previously established value of the parameter is in effect. The previously established value might have been set by an IMS default, the SB Initialization exit routine, PSBGEN, or preceding SBPARM statements.

ACTIV=

Specifies if and when SB should be activated.

NO

Specifies that SB is not activated. NO is the default; however, the default might have been changed by what is specified in the SB Initialization exit routine, PSBGEN, or a preceding SBPARM control statement.

COND

Specifies that IMS monitors the I/O reference patterns of PCBs and eventually activates SB if analysis shows the I/O reference pattern is sequential, and the rate of I/O activity is high enough.

It is recommended that you specify COND for batch and BMP programs that sometimes or always process a database sequentially. SB is not recommended for short-running MPP, Fast Path, or CICS programs because of the overhead associated with initializing SB for each program execution.

If ACTIV= is not specified on an SBPARM control statement, IMS uses the previously established value of the ACTIV parameter. Therefore, omitting ACTIV lets you override the BUFSETS parameter (see below) without inadvertently changing SB PSBGEN specifications.

BUFSETS=4 | nn

Specifies how many buffer sets are to be allocated for a given SB buffer pool. You can specify a value from 1 through 25. The default is 4. (The default can be changed by the SB Initialization exit routine.)

Each SB buffer pool consists of *n* buffer sets. Each buffer set has ten buffers. Each buffer is large enough to hold one OSAM data set block.

You must specify a value greater than 1 if SB is to use overlapped I/O.

A well-organized database might only require a BUFSETS value of 2 for efficient sequential buffering. Less well-organized databases, on the other hand, might require a BUFSETS value of 6 or more. One indicator of database organization can be found in the optional //DFSSTAT reports. If, for a sequential program, these reports show a low percentage of random I/O operations, this can indicate a well-organized database. Conversely, a high percentage of random I/O can indicate a poorly organized database. In either case, the //DFSSTAT reports can help you adjust the BUFSETS value to improve buffering performance.

Qualification Parameters

Qualification parameters specify the I/O operations to which the SBPARM statement applies. You can associate SBPARM statements with a database, PCB, ddname, or PSB.

You can specify any combination of qualification parameters. If you specify more than one qualification parameter, the SBPARM statement applies only to I/O operations that satisfy all qualification parameters. If you do not specify any qualification parameters, the SBPARM statement applies to all I/O operations.

DB=

Specifies the DBD name (as specified in the PCB macro of PSBGEN) of those PCBs to which the SBPARM statement applies.

If multiple PCBs have the same DBD name, and you want different specifications for different PCBs, use the PCB= keyword below. If the PCB refers to multiple database data sets and you want different specifications for these data sets, use the DD= keyword below.

SBPARM Control Statement

If no DB= keyword is provided, the SBPARM statement applies to all I/O operations that satisfy the other qualification keywords.

PCB=

Specifies the label coded on the database PCB during PSBGEN. Use this keyword to distinguish between multiple PCBs with the same DBD name.

If no PCB= keyword is provided, the SBPARM statement applies to all I/O operations that satisfy the other qualification keywords.

DD=

Specifies the ddname of the database data set to which the SBPARM statement applies. Use this keyword to distinguish between multiple data sets to which the PCB refers.

If no DD= keyword is provided, the SBPARM statement applies to all I/O operations that satisfy the other qualification keywords.

PSB=

Specifies the name of the PSB to which the SBPARM statement applies.

If no PSB=keyword is provided, the SBPARM statement applies to all I/O operations that satisfy the other qualification keywords.

Use of the PSB= keyword is strongly recommended for message regions. Normally, SB should not be activated for all programs running in the message region; rather, SB should be restricted to a very small number (if any) of these programs. If SB is to be used for more than one program, you need to provide one SBPARM statement for each program and code the PSBname of the program on the PSB= keyword.

PSB= is also useful if you decide to centralize all SB control statements in one common SB control statement file. In this case, PSB= identifies for each SBPARM statement, the PSB or program to which it applies.

SBPARM examples

Examples of the SBPARM control statement follow:

```
SBPARM ACTIV=COND /*Example 1
SBPARM ACTIV=COND,DB=SKILLDB /*Example 2
SBPARM ACTIV=COND,DB=SKILLDB,PCB=LABEL1 /*Example 3
SBPARM ACTIV=COND,DB=SKILLDB,BUFSETS=6 /*Example 4
SBPARM BUFSETS=6 /*Example 5
SBPARM ACTIV=NO /*Example 6
```

Example 1: SBPARM requests conditional activation of SB for all I/O operations. No default values for SB parameters are overridden.

Example 2: SBPARM requests conditional activation of SB for I/O operations associated with all PCBs coded with DBDNAME=SKILLDB during PSBGEN. No default values for SB parameters are overridden.

Example 3: SBPARM requests conditional activation of SB for all PCBs that satisfy both of the following conditions:

- DBDNAME=SKILLDB coded during PSBGEN
- The PCB label coded as LABEL1 during PSBGEN

Example 4: SBPARM requests conditional activation of SB for all PCBs coded with DBDNAME=SKILLDB during PSBGEN.

SBPARM Control Statement

The BUFSETS parameter is set to 6 (instead of 4, which is the default). This increase in BUFSETS value can improve buffering efficiency if the database is poorly organized.

Example 5: SBPARM requests that the default values of BUFSETS be overridden. Increasing the value of BUFSETS to 6 (rather than using the default, which is 4) can improve buffering efficiency if the database is poorly organized. In this example, ACTIV has not been coded. As a result, the SBPARM statement does not change the previously established set of I/O operations being buffered by SB.

Example 6: SBPARM requests that SB not be used. The request applies to all I/O operations.

SB Image Capture (SBIC) Control Statement

The SBIC control statement captures all internal calls to the SB buffer handler. The captured information is put in the X'5E' log record.

The X'5E' log records can be used as input to the SB test program (DFSSBHD0).

Related Reading: For detailed information on the SB test program, see *IMS/ESA Utilities Reference: Database Manager*.

The format of the SBIC control statement is as follows:

▶▶—SBIC—————▶▶

SB Compare Option (SBCO) Control Statement

The SBCO control statement asks the SB buffer handler to perform a self check to determine if SB is putting incorrect block images in the OSAM buffers, and if so, to provide problem determination information.

Problem determination information is provided in the form of a snap. In the IMS online environment, this snap is written to the IMS log. In the IMS batch environment, you can request that the snap be written to any of the following:

- The data set specified in the DFSDDLTO //PRINTDD DD statement
- A user-specified data set
- The IMS log

See “Snap Destination (SNAPDEST) Control Statement” on page 390 for a description of how to request these snap destinations. In the batch environment, the default snap destination is the data set specified in the //DFSSNAP DD statement.

Related Reading: For detailed information on the SBCO control statement, see *IMS/ESA Diagnosis Guide and Reference*.

The format of the SBCO control statement is as follows:

▶▶—SBCO—————▶▶

SBSNAP Control Statement

SB Snap (SBSNAP) Control Statement

The SBSNAP control statement snaps SB control blocks after each internal call of the OSAM buffer handler to the SB buffer handler. As an option, you can request that, for each internal SB call, the following additional information be snapped:

- The OSAM database block returned by the SB buffer handler to the OSAM buffer handler
- The SB buffers for the current data set group

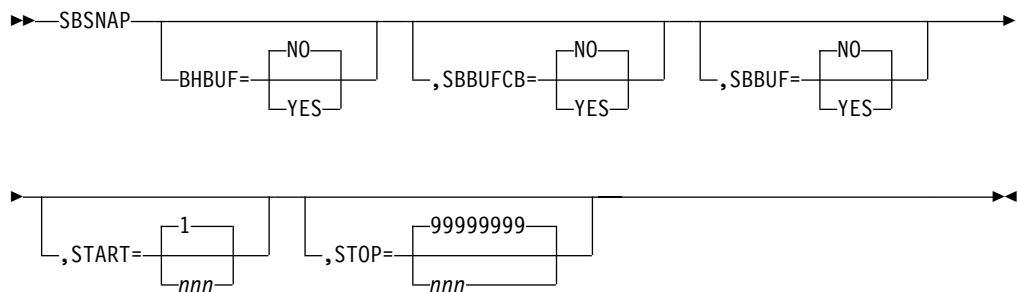
In IMS online environment, this snap is written to the IMS log. In the IMS batch environment, you can request that the snap be written to any of the following:

- The data set specified in the DFSDDLTO //PRINTDD DD statement
- A user-specified data set
- The IMS log

See the section “Snap Destination (SNAPDEST) Control Statement” on page 390 for a description of how to request these snap destinations. In the batch environment, the default snap destination is the data set specified in the //DFSSNAP DD statement.

All SBSNAP parameters are optional.

The format of the SBSNAP control statement is as follows:



If you specify multiple SBSNAP control statements, the n -th control statement overrides the m th control statement (where $n > m$).

Related Reading: For additional information on the SBSNAP control statement, see *IMS/ESA Diagnosis Guide and Reference*.

BHBUF=

Specifies whether (YES) or not (NO) the OSAM buffer handler buffer is to be included in the snap produced at the end of each internal call to the SB buffer handler. The default is NO.

SBBUFCB=

Specifies whether (YES) or not (NO) the buffer control blocks of the SB buffer handler are to be included in the snap produced at the end of each internal call to the SB buffer handler. The default is NO.

SBBUF=

Specifies whether (YES) or not (NO) the buffers of the SB buffer handler are to be included in the snap produced at the end of each internal call to the SB buffer handler. The default is NO.

START=

Specifies that the snap is to start on the *n*th call to the SB buffer handler (instead of the first call). The default is 1.

The START= and the STOP= parameters are useful if you want to limit snapping to a specific period of time during which an application is running. The SBSNAP option often creates a large amount of output.

STOP=

Specifies that the snap is to be stopped on the *m*th call to the SB buffer handler (instead of the last call).

The default is that the snap is stopped after the last call.

SB Evaluation Snap (SBESNAP) Control Statement

The SBESNAP control statement snaps SB control blocks at the end of each periodic evaluation of the buffering process. The information in this snap can help you understand why IMS did or did not use SB.

In the IMS online environment, the snap is written to the IMS log. In the IMS batch environment, you can request that the snap be written to any of the following:

- The data set specified in the DFSDDLTO //PRINTDD DD statement
- A user-specified data set
- The IMS log

See “Snap Destination (SNAPDEST) Control Statement” on page 390 for a description of how to request these snap destinations. In the batch environment, the default snap destination is the data set specified in the //DFSSNAP DD statement.

All SBESNAP parameters are optional.

The format of the SBESNAP control statement is shown below.



If you specify multiple SBESNAP control statements, the *n*th control statement overrides the *m*th control statement (where $n > m$).

Related Reading: For additional information on the SBESNAP control statement, see *IMS/ESA Diagnosis Guide and Reference*.

EVAL=

NEG requests a snap only when a periodical evaluation results in a decision not to use SB.

POS requests a snap only when a periodical evaluation results in a decision to use SB.

ALL requests a snap whenever a periodical evaluation of a data set group is made. The default is NEG.

SNAPDEST Control Statement

Snap Destination (SNAPDEST) Control Statement

The SNAPDEST control statement is applicable in batch environments only. SNAPDEST specifies the destination of the snap created by the SBSNAP, SBESNAP, or SBCO options. In the IMS online environment, SNAPDEST is ignored, and snap information is written to the IMS log.

The default destination for snap information in the batch environment is the data set specified in the //DFSSNAP DD statement.

If you specify an invalid destination in the SNAPDEST control statement, IMS tries to write the snap to the IMS log.

The format of the SNAPDEST control statement is as follows:



The destination specified in the SNAPDEST control statement can be any one of the following:

DFSSNAP

Specifies that snap information is to be put in the data set associated with the //DFSSNAP DD statement.

PRINTDD

Specifies that snap information is to be put in the data set associated with the //PRINTDD DD statement of the DL/I test program DFSDDL0. This data set also contains information created by DFSDDL0.

ddname

Specifies that snap information is to be put in the data set with the specified ddname. The DD statement for this data set must conform to MVS requirements for snap.

When the snap destination is ddname, DCB attributes are provided by IMS. You should not code them on the DD statement.

For this type of destination, each snap opens and closes the data set. If there are many snaps, this can result in significant overhead.

LOG

Specifies that snap information is to be put in the IMS log.

When snaps are written to the log, the log record codes are as follows:

X'67ED'- snap created by the SBESNAP option

X'67EE'- snap created by the SBSNAP option

X'67EF'- snap created by the SBCO option

You can select and print these log records using the File Select and Formatting Print utility (DFSERA10) and the Log Type X'67' Record Format and Print Module (DFSERA30).

Related Reading: Both of these programs are described in *IMS/ESA Utilities Reference: System*.

Specifying High-Speed Sequential Processing Control Statements

This section describes the syntax, keywords, and values of the two High-Speed Sequential Processing (HSSP) control statements: SETO and SETR.

HSSP control statements allow you to:

- Set up the environment in which you process a selected PCB with HSSP
- Create an image copy of a designated DEDB area, thereby reducing the amount of logged data
- Restrict access of a specific HSSP or non-HSSP application program to only designated DEDB areas

HSSO, HSSR, and HSSD control blocks are built from SETO and SETR statements. These control blocks—specifically those that represent image copy datasets and those that are used for UOW locking—are formatted for offline dumps. These control statements are in the DFSCCTL data set.

Syntax of HSSP Control Statements

Each HSSP control statement must begin on a new line. IMS considers the first nonblank character the start of the statement. You must leave one or more blanks following the statement name. Keywords follow the statement name. Keywords must be followed immediately by an equal sign (=) and a value. If you specify multiple keywords, separate them with commas. If you specify multiple values, enclose them in parentheses.

A blank or a closing parenthesis signals the end of a statement. You can continue HSSP control statements on multiple records. Use a comma at the end of a line to indicate continuation.

An asterisk (*) in the first position on a record indicates a comment. Elsewhere on a record, comments begin with a slash and asterisk (/ *). You can include comments on each line with at least one blank between the keyword, value, or comma and the comment.

Set Options (SETO) Control Statement

The SETO control statement allows you to specify the options in processing a PCB with HSSP. With SETO, you can:

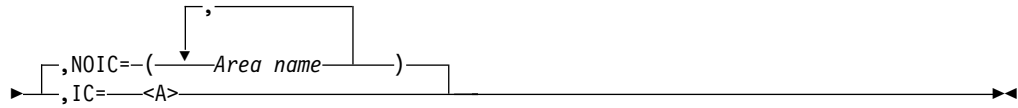
- Deactivate the HSSP option for a particular PCB
- Make an image copy of an updated area
- Specify the processing options in case of image copy failure

IMS interprets SETO when a region is started. After the region is active, you cannot alter the options.

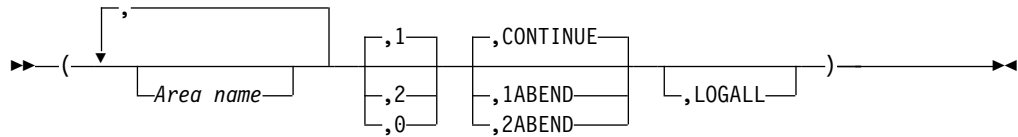
The format of the SETO control statement is:

▶▶—SETO—DB=*dbname*, PCB=*label*—,NOPROCH—▶▶

Specifying High-Speed Sequential Processing



<A> is



Keyword Parameters

DB=

Specifies the name of the database to which SETO applies. Because HSSP is only applicable to DEDBs, only the names of these DEDBs are valid.

PCB=

Specifies a one- to eight- character label that identifies the PCB to which SETO applies. It must be identical to the label on the associated DEDB PCB. If you use the same PCB label more than once for the same database name, only the first SETO statement with that label is used.

NOPROCH

Specify NOPROCH if you want to deactivate the HSSP option that was initialized in PSBGEN (PROCOPT=H). If you specify this keyword, PROCOPT=P is set, and the IC option is ignored. You can only specify **one** HSSP PCB per database per PSB. If more than one HSSP PCB points to the same database, you should use the NOPROCH keyword to deactivate additional PCBs before the SETO statements are processed. If more than one HSSP PCB exists after the SETO statements have been processed, the region is not scheduled.

NOIC=

Specifies that no image copy should be made for all or some of the areas listed in DBDGEN. If you specify NOIC= without any area names, no image copy is made for any area in the database. You can use the option NOIC= with area names several times per SETO statement; however, you can specify a particular area only once. NOIC is the default if you do not specify IC=.

IC=

Specifies that an image copy should be made of all or some of the areas listed in DBDGEN. Allocation information about the image copy data sets is obtained from DBRC.

Values: The following values control the use of HSSP:

Area name

Lists the names of the areas to be image copied. If you need the areas to be processed in a specific order (other than the order in the DBDGEN), you must list them in the desired order and separate them with a comma. If the sequence of areas required is the same sequence as that in the DBDGEN, you only need to list the first and the last area names of the series and

Specifying High-Speed Sequential Processing

separate them with a hyphen. You can specify an area name only once. For example, *A4,A1-A3,A5-A8* is valid, but *A4,A1-A8* is invalid, because *A4* is repeated.

If you specify *IC=* without any area names, an image copy of all areas in the database occurs. You can specify *IC=* without area names only once per *SETO* statement. You can use the option *IC=* with area names several times per *SETO* statement; however, you can specify an area only once.

Commas and parentheses are optional if you specify only one parameter on the *IC=* parameter. You must register the area in *DBRC* in order for image copying to take place.

Related Reading: See *IMS/ESA Utilities Reference: Database Manager* for more information.

- 1 Specifies that one image copy data set should be made. The default is one.
- 2 Specifies that two image copy data sets should be made. If you specify *IC=2*, two image copy data sets must be registered with *DBRC* for an area; otherwise, no image copy is done for that area.
- 0 Specifies that no image copy data sets should be made. *IC=0* is the same as specifying *NOIC*.

CONTINUE | 1ABEND | 2ABEND

These keywords describe the action a program should take in the event of an image copy failure.

CONTINUE specifies that the program continues processing if the image copy option cannot be performed. Under this circumstance, messages are sent to the *MTO* console and the job log (*WTP*). **CONTINUE** can be abbreviated as *C*.

1ABEND specifies that if image copying cannot be completed for one data set, the program must abend. **1ABEND** can be abbreviated as *1A*.

2ABEND specifies that if image copying cannot be completed for two data sets, the program must abend. If you request *IC=2* and one of the data sets is still usable, *HSSP* continues to write to it. **2ABEND** can be abbreviated as *2A*.

LOGALL

Specifies that *X'5950'* log records are to be logged for *DEDB* updates in an *HSSP* environment. If **LOGALL** is not specified, only the *X'5947'* log record is logged.

In an *RSR* environment, only *X'5950'* log records are logged, regardless of the specification of the **LOGALL** parameter.

If you specify only one parameter on the *IC=* keyword, you do not need to include the parentheses or the commas. Trailing commas for those parameters you do not include are also not required. For example, *IC=1* is equivalent to *IC=(,1,,)*, and *IC=1,LOGALL* is equivalent to *IC=(,1,,LOGALL)*.

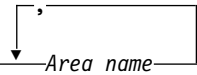
Set Range (SETR) Control Statement

The **SETR** control statement specifies the processing range of *PCBs* to a database during scheduling of an application program. If a *PSB* has several *PCBs* pointing to

Specifying High-Speed Sequential Processing

the same database, you can restrict the access of each PCB to that database. Each program can only access data in the DEDB within the range defined.

The format of the SETR control statement is:

```
▶▶—SETR—DB=dbname,PCB=label,AL=(Area name)—▶▶
```

Keyword Parameters

DB=

Specifies the name of the database to which the SETR statement applies.

PCB=

Specifies a one- to eight-character label that identifies the PCB to which the SETR statement applies. This label must be identical to the label on the associated DEDB PCB. If you use the same PCB label more than once for the same database name, only the first SETR statement with that label is used.

AL=

Specifies the areas to which the PCB has access. The SETR area list does not have to match the SETO area list. (The SETR area list determines which areas can be processed. The SETO area list indicates which of the processed areas must have an image copy.) If you specify the SETR statement without a matching SETO statement (that is, the same DB= and PCB= values), the SETR statement processes the specified area list (AL=) values according to the PROCOPT option in PSBGEN.

HSSP Control Statement Examples

The following are examples of SETO and SETR statements in the DFSCCTL data set with their associated PCBs.

Sample PSBGEN:

```
L1 PCB TYPE=DB,DBDNAME=DEDDB1,PROCOPT=A
L2 PCB TYPE=DB,DBDNAME=DEDDB2,PROCOPT=HI
L3 PCB TYPE=DB,DBDNAME=DEDDB3,PROCOPT=HRD
L4 PCB TYPE=DB,DBDNAME=DEDDB4,PROCOPT=HA
L5 PCB TYPE=DB,DBDNAME=DEDDB5,PROCOPT=HG
```

Sample DFSCCTL set option statements:

```
SETO DB=DEDDB1,PCB=L1,IC=1 /* Example 1
SETO DB=DEDDB2,PCB=L2,IC=(1,CONTINUE) /* Example 2
SETO DB=DEDDB3,PCB=L3,IC=(A1,A3-A6,2,1A) /* Example 3
SETR DB=DEDDB3,PCB=L3,AL=(A1-A5)
SETO DB=DEDDB4,PCB=L4,IC=(2,2ABEND) /* Example 4
SETO DB=DEDDB5,PCP=L5,NOIC=(A4-A5) /* Example 5
SETO DB=DEDDB5,PCB=L5,IC=1,LOGALL
```

Example 1

This SETO statement matches the PCB with label L1. However, the PROCOPT=A defines it as a non-HSSP PCB. The SETO statement is ignored.

Example 2

Specifying High-Speed Sequential Processing

This SETO statement matches the PCB with label L2. The PROCOPT=HI defines it as an HSSP PCB.

- IC=** Indicates a list of values for the image copy option.
- 1** Requests one image copy of all referenced areas. A referenced area is an area for which a DL/I call has been issued.
- CONTINUE** Indicates that processing should continue if image copying cannot be performed.

Example 3

This SETO statement matches the PCB with label L3.

- A1,A3–A6** Requests an image copy of each indicated area when referenced by the application.
- 2** Requests two image copies of all referenced areas.
- 1ABEND** Requests that the program must abend if one image copy fails.

This SETR statement matches the PCB with label L3. The AL=(A1=A5) sets the range of areas to which PCB L3 has access.

Example 4

This SETO statement matches the PCB with label L4.

- 2** Requests two image copies of all referenced areas
- 2ABEND** Requests that the program must abend if both image copies fail. If one image copy data set is usable, the program continues to write on it.

Example 5

This SETO statement matches the PCB with label L5.

- NOIC=** Excludes the areas for which HSSP image copying is to be done.
- A4–A5** Indicates the areas for which no image copies are requested.

Example 6

This SETO statement matches the PCB with label L5. The PROCOPT=HI defines it as an HSSP PCB..

- IC=** Indicates a list of values for the image copy option.
- 1** Requests one image copy of all referenced areas. A referenced area is an area for which a DL/I call has been issued.
- LOGALL** Indicates that updates are logged in X'5950' log records.

Supporting CCTL Users with DBCTL Databases

DBCTL is an IMS facility that allows access to DL/I full-function databases and data entry databases (DEDBs) from a subsystem called a coordinator controller subsystem (CCTL). A CCTL is the transaction management subsystem that communicates with the database resource adapter (DRA).

Supporting CCTL Users with DBCTL

The following sections describe the installation tasks required of the CCTL user who will use the DBCTL interface, the DRA.

Related Reading: CCTL programming requirements for using the DRA interface appear in “Using the Database Resource Adapter (DRA)” in the *IMS/ESA Application Programming: Database Manager*. Specific installation concerns related to CICS as a CCTL are covered in “Chapter 9. Accessing IMS Databases with CICS” on page 415.

The diagram below illustrates the relationship of the DRA interface, CCTL and DBCTL.

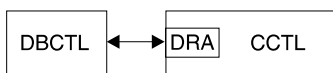


Figure 12. DRA Interface to DBCTL

As part of the DBCTL generation process, you should consider CCTL requirements such as database, PSB, or buffer pools. For sample input for DBCTL stage 1 generation, see “IMS DBCTL Environment” on page 458.

Preparing a CCTL

Two steps you must take to prepare a CCTL for enabling a DRA environment are:

1. The DRA startup/router routine (DFSPRRC0) must be in a CCTL load library. The routine can be copied from the IMS.RESLIB which is built by the IMS generation process. Or, the IMS.RESLIB can be concatenated in the CCTL step library.
2. The DFSPZPxx load module must be in a CCTL load library; it is the DRA startup table. The xx is the startup table name suffix that the CCTL specifies on the DRA initialization request. The source code for DFSPZPxx is DFSPZP00 in the IMS distribution library, DLIB. After all modifications have been made to DFSPZPxx, it is assembled using IMS DLIBS. A default load module, DFSPZP00, is in the IMS.RESLIB. It has default values for all but two of the required DRA initialization parameters, all of which can be overridden on the initialization (INIT) request itself.

The rest of the DRA code must reside in a load library that is dynamically allocated by DFSPRRC0. The DDNAME and DSNAME of this load library are specified in the startup table or initialization request. The default DSNAME is IMS.RESLIB because all DRA code resides here throughout the IMS generation process.

The DRA Startup Table

Sample DFSPZP00 Source Code

The DRA startup table, DFSPZPxx, is created by assembling the DFSPZPxx module. The following code is the actual source code for DFSPZP00. To define other variations of DFSPZPxx, modify this code through the DFSPRP macro. Specify the DRA parameters as keywords on the DFSPRP macro. The keywords and their descriptions appear following this sample:

```
TITLE 'DATABASE RESOURCE ADAPTER STARTUP PARAMETER TABLE'
DFSPZP00 CSECT
*****
*                                                                 *
*  MODULE NAME: DFSPZP00                                       *
*                                                                 *
*                                                                 *
```

Supporting CCTL Users with DBCTL

```

*   DESCRIPTIVE NAME: DATABASE RESOURCE ADAPTER (DRA)           *
*   STARTUP PARAMETER TABLE.                                   *
*   *                                                           *
*   FUNCTION: TO PROVIDE THE VARIOUS DEFINITIONAL PARAMETERS  *
*   FOR THE COORDINATOR CONTROL REGION. THIS                   *
*   MODULE MAY BE ASSEMBLE BY A USER SPECIFYING               *
*   THEIR PARTICULAR NAMES, ETC. AND LINKEDITED                *
*   INTO THE USER RESLIB AS DFSPZPXX . WHERE XX                *
*   IS EITHER 00 FOR THE DEFAULT, OR ANY OTHER ALPHA-        *
*   NUMERIC CHARACTERS.                                        *
*   *                                                           *
*   KEYWORDS FOR THE DFSPRP MACRO:                               *
*   DSECT=NO-A DSECT STATEMENT FOR PRP WILL NOT BE            *
*   GENERATED (LABEL DFSPRP WILL BE ON DS 00).                 *
*   *                                                           *
*   FUNCLV= DEFAULT (1). ADAPTER FUNCTIONAL LEVEL.            *
*   *                                                           *
*   DDNAME= 1 TO 8 CHARACTER DD NAME TO BE USED WITH          *
*   DYNAMIC ALLOCATION OF THE DBCTL RESLIB.                     *
*   DEFAULT (CCTLDD).                                          *
*   *                                                           *
*   DSNAME= 1 TO 44 CHARACTER DATASET NAME OF THE             *
*   DBCTL RESLIB.                                             *
*   DEFAULT (IMS.RESLIB).                                       *
*   *                                                           *
*   DBCTLID=XXXX-NAME OF THE DBCTL REGION                      *
*   DEFAULT = SYS1                                             *
*   *                                                           *
*   USERID=XXXXXXXX-NAME OF THE USER REGION                   *
*   *                                                           *
*   MINTHRD=XXX-MINIMUM NUMBER OF THREADS TO BE              *
*   AVAILABLE (MAXIMUM NUMBER IS 255)                          *
*   DEFAULT = 1                                               *
*   *                                                           *
*   MAXTHRD=XXX-MAXIMUM NUMBER OF THREADS TO BE              *
*   AVAILABLE (MAXIMUM NUMBER IS 255)                          *
*   DEFAULT = 1                                               *
*   *                                                           *
*   TIMER=XX-IDENTIFY TIMER VALUE IN SECONDS (DEFAULT 60)    *
*   *                                                           *
*   FPBUF=XXX-NUMBER OF FAST PATH BUFFERS TO BE ALLOCATED    *
*   AND FIXED PER THREAD (DEFAULT 00)                          *
*   *                                                           *
*   FPBOF=XXX-NUMBER OF FAST PATH OVERFLOW BUFFERS TO BE     *
*   ALLOCATED PER THREAD (DEFAULT 00)                          *
*   *                                                           *
*   SOD=X-OUTPUT CLASS TO BE USED FOR SNAP DUMP OF            *
*   ABNORMAL THREAD TERMINATIONS (DEFAULT A)                  *
*   *                                                           *
*   TIMEOUT=XXX-DRATERM TIMEOUT VALUE IN SECONDS(DEFULT 60)*
*   *                                                           *
*   CNBA=XXXX TOTAL FP NBA BUFFERS FOR CCTL                    *
*   *                                                           *
*   *                                                           *
*   AGN=XXXXXXXX-1 to 8 CHARACTER APPLICATION GROUP NAME      *
*   *                                                           *
*   LOCATION: PRIVATE STORAGE, USER KEY                        *
*   *                                                           *
*   THIS MODULE CONTAINS NO EXECUTABLE CODE.                  *
*   *                                                           *
*   *                                                           *
*****
EJECT
DFSPRP DSECT=NO
END

```

Supporting CCTL Users with DBCTL

DFSPRP Macro Keywords

The following information describes DRA table parameters specified through the DFSPRP macro in DFSPZP00.

AGN= Is a one- to eight-character application group name that is used as part of the DBCTL security function.

Related Reading: See *IMS/ESA Administration Guide: System* for more information.

CNBA=

Is the total number of Fast Path NBA buffers for the CCTL's use.

Related Reading :For a description of Fast Path DEDB buffer use, see *IMS/ESA Administration Guide: System*.

DBCTLID=

Is the four-character name of the DBCTL region. This is the same as the IMSID parameter in the DBC procedure. The default name is SYS1. For more information on the DBC procedure, see "DBC Procedure" on page 195.

DDNAME=

Is a one- to eight-character ddname to be used with the dynamic allocation of the DBCTL RESLIB library. The default ddname is CCTLDD. This library must contain the DRA modules.

DSNAME=

Is a one- to 44-character data set name of the DBCTL RESLIB library. This library must contain the DRA modules and must be MVS authorized. The default dsname is IMS.RESLIB.

FPBOF=

Is the number of Fast Path DEDB overflow buffers to be allocated per thread. The default is 00.

Related Reading: For a description of Fast Path DEDB buffer use, see *IMS/ESA Administration Guide: System*.

FPBUF=

Is the number of Fast Path DEDB buffers to be allocated and fixed per thread. The default is 00.

Related Reading: For a description of Fast Path DEDB buffer use, see *IMS/ESA Administration Guide: System*.

FUNCLV=

Specifies what level of the DRA the CCTL supports. FUNCLV=1 the default. No other value is possible.

MAXTHRD=

Is the maximum number of DRA thread TCBs to be available at any given time. The maximum number is 255. The default is 1.

MINTHRD=

Is the minimum number of DRA thread TCBs to be available at any given time. The maximum number is 255. The default is 1.

SOD= Is the output class to be used for a SNAP DUMP of abnormal thread terminations. The default is A.

TIMEOUT=

Is the amount of time (in seconds) a CCTL should wait for the successful completion of a DRA TERM request. This value should be specified only if

Supporting CCTL Users with DBCTL

the CCTL is coded to use it. This value is returned to the CCTL upon completion of an INIT request. The default is 60 seconds.

TIMER=

Is the amount of time (in seconds) between attempts of the DRA to identify itself to DBCTL during an INIT request. The default is 60 seconds.

USERID=

Is an eight-character name of the CCTL region.

Enabling ETO Support for ACF/VTAM Terminals

To use ETO, at least one user descriptor and one logon descriptor must exist. If at least one of each of these descriptors does not exist when you enable ETO, IMS abends with U0015 issuing message DFS3652.

Related Reading: For information on exit routines associated with ETO, refer to *IMS/ESA Customization Guide*. For information on what is required to enable ETO in your installation, refer to the ETOFEAT= keyword in the IMSCTRL macro (“IMSCTRL Macro” on page 82) and the ETO= keyword in the IMS procedure (“IMS Procedure” on page 221).

ETO Descriptors

When you are enabling ETO support in your installation, you can optionally request that IMS produce ETO descriptors during system definition. You do this with the ETOFEAT keyword in the IMSCTRL macro (“IMSCTRL Macro” on page 82). Descriptors contain information about terminal devices and users that is required to establish dynamic sessions. Four types of descriptors are:

Logon

MFS Device

MSC

User

Descriptors that are created during system definition are placed in member DFSDSCMx, which resides in IMS.PROCLIB. If you perform subsequent system definitions of the same stage 1 input deck, the DFSDSCMx member are overwritten. If you use TSO or an MVS utility to create descriptors, you can place the descriptors in member DFSDSCTx to avoid loss in the event that member DFSDSCMx is replaced. You can also use TSO or an MVS utility to update descriptors in DFSDSCMx that were created during system definition.

Descriptor Format

The suffix for DFSDSCMx will always be the nucleus suffix. The default suffix for DFSDSCTy defaults to 0; however, you can specify it with the DSCT= JCL parameter. Each descriptor statement must be 80 characters in length. All statements are translated to upper case. The basic format of each of the descriptor types is shown in Table 30.

Related Reading: For more information on descriptor formats, refer to *IMS/ESA Administration Guide: Transaction Manager*.

Enabling ETO Support for ACF/VTAM

Table 30. Descriptor Type Formats

Column Number	Contains	Considerations
01	The descriptor type indicator: L (logon) M (MSC) U (user) D (MFS device)	If an asterisk (*) is in column 01, the record is considered a comment record and is ignored.
03-10	The descriptor name	<ul style="list-style-type: none">• All descriptor names must be 1-8 alphanumeric characters• The first character for logon and user descriptors must be alphabetic (A-Z, #, \$, or @)• Duplicate descriptor names are allowed if they are grouped together, because more parameters might be required than can exist in an 80-byte record• No more than 50 records (excluding comments) are permitted for each individual descriptor• For MSC descriptors, this column contains the link name
12-72	Parameters for each descriptor	<ul style="list-style-type: none">• Parameters vary by type• They are delineated by blanks• They are typically in the same format as the equivalent parameters specified in the stage 1 input deck
73-80	Can contain sequence numbers	IMS ignores these columns

Rules for Creating Descriptors

You should be aware of the following general guidelines when creating descriptor input streams.

Keywords:

- Separate one keyword or parameter set from another with one or more blanks.
- Do not include imbedded blanks within a keyword and its parameters.
- Separate a keyword from its parameters with an equal sign (=).
- Do not abbreviate keywords.

You can continue a keyword and its parameter set to the next statement if no intervening blanks appear at the end of the first statement or at the beginning of the parameters of the next statement. A continued statement still has the same descriptor type and name in columns 1-10; the continued specification begins in column 12. If you specify keywords, they must be accompanied by parameters. Keywords followed by blanks or commas are invalid.

Parentheses: Parentheses are optional if they follow a keyword that supports multiple parameters and only if the first parameter is specified. Otherwise, they are required.

Parameters:

- Separate multiple parameters for single keywords with commas.
- Specify at least one parameter for each keyword.

Enabling ETO Support for ACF/VTAM

If the keyword has positional parameters, you can use commas when parameters are specified after a missing positional parameter. Following is an example of valid use:

```
LTERM=(ABC,,1)
```

This example is valid because LTERM has positional parameters.

The following example is invalid:

```
LTERM=(ABC,,1,)  
OPTIONS=(TRANSRESP,)  
OPTIONS=(,SYSINFO)
```

The LTERM example is invalid, because the comma must be followed by a parameter. The OPTIONS examples are invalid, because the parameters for the OPTIONS keyword are not positional.

Descriptor Overrides: If a descriptor is defined in both DFSDSCTy and DFSDSCMx, the descriptor in member DFSDSCTy overrides the descriptor in member DFSDSCMx. No message is issued. If one descriptor is defined more than once in either member, error message DFS3661 is issued. IMS uses the last descriptor defined. If the same descriptor is defined once in member DFSDSCTy and one or more times in member DFSDSCMx, the descriptor in member DFSDSCTy is used, and no error message is issued.

Logon Descriptors

When you request that ETO descriptors be created during system definition, a logon descriptor is created for each VTAM TYPE and TERMINAL macro set. Exceptions to this are:

- Any terminal defined as a primary or secondary MTO
- Any LUTYPE6 terminal defined as the XRF ISC link

ETO logon descriptors are not produced for these terminals.

Also produced during system definition is a default logon descriptor, which, reflects the TYPE and TERMINAL macro statements that define the unique characteristics of the largest number of terminals of that type. Terminals that cannot be represented by the default logon descriptor have unique logon descriptors created for them. You can keep and use these unique logon descriptors or discard them by:

- Operating these terminals based on the default logon descriptor
- Creating one or more of your own installation default logon descriptors and coding an installation exit routine DFSLGNX0, to select the required descriptor for logging on to a terminal

The system definition process generates up to 37 default logon descriptors for each device type. ETO descriptors are not generated during large system generation (LGEN). The suffix for each descriptor defaults to 0; the last character of the descriptor name ensures that the name is unique. This character is a blank for the most common descriptors, and then 0 through 9, and A through Z, for a total of 37 possible descriptors.

Terminal definitions that do not match the 37 most common terminal definitions generate comment statements with an asterisk in the first position and the rest of the statement is shifted over one position.

Enabling ETO Support for ACF/VTAM

All user descriptors created during system definition are generated as comment statements to avoid getting error messages caused by a statically defined LTERM name.

Typically, one generic logon descriptor exists per unique LU type and option configuration. After the descriptor is created, the same descriptor can support any number of logons for the same LU type and option configuration.

Logon Descriptor Format

The format of a logon descriptor is

```
L Descriptor name Parm(1) Parm(2) Parm(3)
L Descriptor name Parm(4) Parm(5) Parm(n)
```

The parameters that are required for a descriptor cannot fit into one record. You might need to use multiple records with the same descriptor name to define your descriptor.

Related Reading: Refer to *IMS/ESA Administration Guide: Transaction Manager* for more information on descriptor formats.

Logon Descriptor Parameters

Because logon descriptors are built from equivalent TYPE and TERMINAL macros, most definitions of logon descriptor parameters match the parameters found on those macro statements. Parameters that are not defined below are **not** supported for logon descriptors.

descriptor type

Specifies that the descriptor is logon (L).

descriptor name

Specifies the name of the logon descriptor name. Default names include DFS3270, DFS327P, DFSFIN, DFSSLU1, DFSSLU2, DFSSLUP, DFSLU61, and DFSNTO.

ALOT=

Specifies the autologoff time in minutes. Valid values are 0 and from 10 to 1440. If the ALOT value is not specified, the value from the JCL member is used except for FINANCE, SLU P, and ISC. If ALOT is not specified on the logon descriptor or overridden by the logon exit (DFSLGNX0) for FINANCE, SLU P, and ISC, a value of 1440 is used (the value from the JCL member is ignored).

- ALOT= 0

The terminal is logged-off immediately when no signon is in effect. This specification is normally used in terminal sessions when the user is signed-on automatically during the logon process. During autologon, signon data can be provided in one of the following ways:

- Signon data supplied via the IMS /OPNDST command
- Signon data supplied by logon userdata (BIND)
- Signon data supplied by logon exit (DFSLGNX0)

There are two modes of operation for using ALOT=0, either of which can be set using the DFSINTX0 User Initialization Exit parameter list.

In default mode, when signon errors are encountered, the session is automatically signed off and then logged off; no message is sent. If you do not supply the DFSINTX0 exit, or you supply the exit and indicate default

Enabling ETO Support for ACF/VTAM

mode for ALOT=0, then signon data must be supplied during the logon process. All of the following error conditions result in automatic logoff:

1. A non-signon, or errors detected during signon or input processing, result in immediate logoff.
2. /SIGNOFF results in immediate logoff .
3. /SIGNON signs off the current user and signs on a new user. However, errors encountered during the signon process, such as detection of an incorrect or expired password, result in immediate logoff.

Restriction: Default mode should not be used for interactive terminal sessions that require a response to the DFS3649 message; these sessions will not wait for input signon and will logoff immediately.

In alternate mode, when signon errors are encountered, the session is automatically signed off, a message is sent and the session is logged off. Signon data can be supplied but is not required. All of the following error conditions result in automatic logoff:

1. A non-signon error detected during input processing results in immediate logoff.
 2. No signon data has been provided by the logon userdata (BIND) or the Logon Exit (DFSLGNX0).
 3. A /SIGNOFF, or errors resulting from a /SIGNON, cause message DFS3649(A) (Signon Required) to be sent, and a fixed ten-minute timer set to wait for a new signon. If no signon occurs during that interval, then the session will be logged off.
- ALOT=(10-1439)
The session is terminated after the specified number of minutes have elapsed without a signed-on user.
 - ALOT=1440
The session is never automatically terminated.

ASOT=

Specifies the autosignoff time, in minutes. Valid values are 0 and from 10 to 1440. You can override this value during signon with the user descriptor. If ASOT is not specified on either the user descriptor, the logon descriptor, or overridden by either the logon (DFSLGNX0) or signon (DFSSGNX0) exits, the value from the JCL member is used except for FINANCE, SLU P, and ISC. If ASOT is not specified on either the user or logon descriptor for FINANCE, SLU P, and ISC, a value of 1440 is used (the value from the JCL member is ignored).

- ASOT=0
The user is signed off immediately when no output is available to be sent. ASOT=0 is normally specified when no IMS input or output is available, or after the last available output message completes.
Restriction: Do not specify ASOT=0 for interactive terminals, for example 3270s or SLU2.
- ASOT=(10-1439)
The user is signed off after the specified number of minutes have elapsed without terminal activity, irrespective of message status.
- ASOT=1440
The user is never automatically signed off.

Enabling ETO Support for ACF/VTAM

BACKUP=

Specifies the control of automatic restart (BTAM) or session switching (VTAM) after takeover. Use only when HSB=YES on the IMSCTRL macro.

This keyword is the same as the BACKUP= keyword in the TYPE macro. See "TYPE Macro" on page 186 for more information on keyword values.

COMPTn=

Where *n* is 1-4, specifies the first through fourth components of this logon descriptor. This keyword is only valid for the following device types: 3270, FINANCE, SLU 1, SLU P, and LU6.1 (ISC). For UNITYPE=FINANCE, these four keywords are the same as the four operands on the TERMINAL macro's COMPT keyword. The parameters for this keyword are positional. See "TERMINAL Macro" on page 140 for more information on this keyword for other device types.

COMPT=PTR1 for UNITYPE=3270, UNIT=3275 should be specified as COMPT2=PTR1.

EDIT=

Specifies the name of a user-supplied physical terminal output and input edit routine for the terminals in this communication description set.

This keyword is the same as the EDIT= keyword in the TYPE macro. See "TYPE Macro" on page 186 for more information on keyword values.

FEAT=

The parameters for this keyword are not positional.

See the description of the FEAT= keyword in "TERMINAL Macro" on page 140 and "TYPE Macro" on page 186 for more information on this keyword.

MODETBL=

Specifies the name of the VTAM logon mode table entry (logon mode name) containing the SNA bind parameters to be used when a session is established for this terminal.

This keyword is the same as the MODETBL= keyword in the TERMINAL and TYPE macros. See "TERMINAL Macro" on page 140 and "TYPE Macro" on page 186 for more information on this keyword.

OPTIONS=

Specifies certain communication options associated with this terminal. Parameters for this keyword are not positional.

Aside from the exceptions noted below, the communications options described in the OPTIONS= keyword of the TYPE and TERMINAL macros are supported for logon descriptors. See "TERMINAL Macro" on page 140 and "TYPE Macro" on page 186 for more information on this keyword.

The following options are not supported for logon descriptors:

- NOSHARE|SHARE
- ACK|OPTACK
- NORESP|FORCRESP|TRANRESP

NORESP|FORCRESP|TRANRESP can be specified on the user descriptor.

Enabling ETO Support for ACF/VTAM

The COPY|NOCOPY option is supported for logon descriptors for dynamic SLU 2 terminals, but not for dynamic non-SNA VTAM 3270s.

OUTBUF=

Specifies the size of the IMS output buffer to be used for workstations.

This keyword is the same as the OUTBUF= keyword on the TERMINAL macro. See “TERMINAL Macro” on page 140 for more information on this keyword.

PTRSIZE=

Specifies the number of print positions of the 3284 or 3286 printer. The default is 120.

If PTRSIZE=IGNORE is specified with FEAT=IGNORE in its DEV statement, MFS always uses the device output format (DOF) when editing output for this device. The existing parameters and control block values, including default parameters, are unchanged.

SEGSIZE=

Specifies segment size. Acceptable values are from 256 to 32000 bytes. The default is 256. For information on calculating segment size, see the description of the SEGSIZE= keyword in “TERMINAL Macro” on page 140.

UNIT=

Specifies the terminal for the previously defined line group.

This keyword is the same as the UNIT= keyword on the TERMINAL macro. See “TERMINAL Macro” on page 140 for more information on this keyword.

UNITYPE=

Specifies the UNITYPE of the terminal. Valid UNITYPEs include 3270, FINANCE, SLUTYPE1, SLUTYPE2, SLUTYPEP, LUTYPE6, or NTO. 3601 is an invalid UNITYPE. If not specified, a default UNITYPE of SLUTYPE2 is used.

Logon Descriptor Example

Following is an example of a logon descriptor:

```
L MKT01LU2 UNITYPE=SLUTYPE2 ALOT=30
```

MFS Device Descriptors

MFS device descriptors are used by the MFS Device Characteristics Table (DCT) utility. This utility updates screen information, for example 3270 screen sizes and feature information, in the DCT and generates new MFS default formats without system generation.

You use MFS device descriptors to define characteristics of terminals added dynamically that have different characteristics from statically defined terminals.

Related Reading: See *IMS/ESA Utilities Reference: Transaction Manager* for more information about how to use the DCT utility.

MFS Device Descriptor Format

The format of an MFS device descriptor is

```
D Descriptor name Parm(1) Parm(2) Parm(3)
```

MFS Device Descriptor Parameters

The three parameters supported for MFS device descriptors are TYPE=, SIZE=, and FEAT=.

Enabling ETO Support for ACF/VTAM

Example of MFS Device Descriptors:

```
Co1      Co1
1        12
D        TYPE=3270-A04 SIZE=(43,80) FEAT=IGNORE
```

See the parameter descriptions in “TERMINAL Macro” on page 140 for more information.

MSC Descriptors

MSC descriptors relate remote NAME macros to MSC links defined during system definition. When you specify with ETOFEAT that descriptors are to be created during system definition, an MSC descriptor is created for each MSNAME macro statement.

MSC Descriptor Format

The format of an MSC descriptor is

```
M Link name name1 name2 namen
```

An MSC descriptor contains a link name rather than a descriptor name. The link name is the name of the retrieved descriptor. The name parameters represent any valid remote LTERM name that can be accessed through a link name. The parameters that are required for a link name cannot fit into one record. You might need to use multiple records with the same link name to define your descriptor.

MSC Descriptor Parameters

The parameters that are supported for MSC descriptors are as follows:

descriptor type

Specifies that the descriptor is MSC (M).

descriptor name

Specifies the linkname from the MSNAME macro.

remote LTERM name

Specifies the name of a logical terminal associated with a physical terminal defined in a remote IMS system.

MSC Descriptor Examples

Following are examples of MSC descriptors:

```
M REMSYS01 REM01AAA REM01BBB REM01CCC REM01DDD REM01EEE REM01FFF
M REMSYS01 REM01GGG REM01HHH REM01III
```

User Descriptors

User descriptors are another form of descriptor created during system definition for ETO support. Three types of user descriptors are:

- DFSUSER
- Node
- Installation-created

The node user descriptors are essentially migration aids, allowing message queues and options associated with a particular terminal to remain unchanged during migration. They do not, however, provide output security.

Related Reading: For information on the format of user descriptors, refer to *IMS/ESA Administration Guide: Transaction Manager*.

Enabling ETO Support for ACF/VTAM

User descriptors are generated from each VTAM TERMINAL or VTAMPOOL SUBPOOL macro statement. Those descriptors created from VTAM TERMINAL macro statements, such as node user descriptors, have the same name as the terminal. Descriptors created from VTAMPOOL SUBPOOL macro statements have the same name as the subpool.

For user descriptors created during IMS system definition from the SUBPOOL macro statement, the response option (TRANSRESP, NORESP, or FORCRESP) cannot be set. This is because the response option is defined on the TERMINAL statement for static definitions. You need to add the appropriate response option for your installation to any user descriptor created from a SUBPOOL statement.

DFSUSER Descriptor

When you request that IMS build ETO descriptors, IMS creates a single default user descriptor, DFSUSER, which defines default user characteristics. The DFSUSER descriptor represents the most common set of user options. With this descriptor, IMS can dynamically create a user and message queue structure for a signon request when no other user descriptor is available. The message queue name is the same as the user ID. You can also use DFSUSER to add queue names and other options through the DFSSGNX0 Signon exit routine.

Related Reading: Refer to *IMS/ESA Customization Guide* for more information on how to use the DFSSGNX0 exit routine.

Recommendation: Use the DFSUSER descriptor for most of your users after your migration to ETO is completed.

Node User Descriptors

Node user descriptors are created during system definition. One is generated for each terminal in the IMS system definition. Node user descriptors are created even if they match DFSUSER options, except for ISC terminals defined for parallel-session support.

Installation-Created Descriptors

Installation-created user descriptors are designed by your installation to meet your criteria. The name of the installation-created user descriptor is the same as the user ID.

User Descriptor Format

The format for a user descriptor, regardless of whether it is created by installation, node, or DFSUSER, is:

```
U username parm1 parm2 parmn
```

U is the descriptor type (USER), and Username is DFSUSER, the user ID, or the node user name from the name field in the TERMINAL or SUBPOOL macro statement supplied by the installation.

parm (1..n) can be any of the following parameters: ASOT=, LTERM=, OPTIONS=, AUTLGN=, AUTLMOD=, AUTLDESC=, AUTLID=

User Descriptor Parameters

Unless specifically noted below, the definition of a given user descriptor parameter is the same as that found for the equivalent NAME ("NAME Macro" on page 127) and SUBPOOL macro statement ("SUBPOOL Macro" on page 139). Also, unless specifically noted, the general rules for creating descriptors apply to the following user descriptors. Parameters not defined below are not supported for user descriptors.

Enabling ETO Support for ACF/VTAM

The following parameters are valid on the user descriptor:

ASOT=a

a is 0 or a value from 10 to 1440. The default comes first from the logon descriptor parameter ASOT. If ASOT is not specified on either the user or logon descriptor, the value from the JCL member is used except for FINANCE, SLU P and ISC. If ASOT is not specified on either the user or logon descriptor for FINANCE, SLU P, and ISC, or overridden by the logon (DFSLGNX0) or (DFSSGNX0) exits, then a value of 1440 is used, the value from the JCL member is ignored.

AUTLDESC=d

d is a one- to eight-byte alphanumeric logon descriptor name, the first character of which is alphabetic (A-Z, \$, #, @). AUTLDESC=d defines the characteristics of the terminal to be autologged on. This parameter is ignored for LU 6.1 devices.

AUTLGN=b

b is a one- to eight-byte alphanumeric LU name for auto logon, the first character of which is alphabetic (A-Z, \$, #, @). Autologon allows IMS to logon and signon your terminal automatically. If you specify the autologon option for a user, the queueing of data to any of the user queues causes IMS to establish a session.

Related Reading: For more information, see *IMS/ESA Administration Guide: Transaction Manager*.

AUTLID=g

g is a one- to eight-byte alphanumeric ISC other system half session qualifier, of which the first character is alphabetic (A-Z, \$, #, @). If the other system is IMS, then this is the name of an ISC user in that system. The AUTLGN parameter must be used with the AUTLID parameter when requesting autologon for an ISC parallel session.

AUTLMOD=e

e is a one- to eight-byte alphanumeric mode table for the autologon terminal.

LTERM=(f,h,i,j)

f, *h*, *i*, and *j* are all positional parameters. *f* is a one- to eight-byte alphanumeric LTERM name. *h* is either ULC or UC; the default is ULC. *i* is the COMPT parameter, and *j* is the ICOMPT parameter. COMPT and ICOMPT are the same as defined by keyword parameters on the NAME macro. If *f* is the only parameter supplied, the parentheses can be omitted.

A maximum of eight queues can be specified for each user descriptor.

If no LTERM keyword is specified, the user descriptor control block structure is built with one queue. The name used as the descriptor name is the default.

The queue names specified on the LTERM parameter keyword for the user descriptors must be unique. That is, two user descriptors cannot be created with the same queue name. If duplicate LTERM names are specified, message DFS3669 is issued.

If the LTERM keyword is specified on the user descriptor, the queue name must also be specified. The DFSUSER descriptor must also follow this convention (for example, with queue name DFSUSER).

Enabling ETO Support for ACF/VTAM

The queue names (and Remote LTERM names) must follow the same naming conventions as the user names because they can, in some circumstances, take on the same name as the user name. The user name has strict naming conventions because it must follow RACF naming conventions. Therefore, these names must be alphanumeric (A-Z, 0-9, \$, #, @), but the first character in the name must not be 0-9.

OPTIONS=j,k

j is a response mode of FORCRESP, TRANRESP, or NORESP. The default response mode is dependent upon the device type. If the response mode is not explicitly set in the user descriptor, the default cannot be set until signon, when the device type is known. The default setting for each device type, during signon, is indicated by an X in the corresponding column of Table 31.

Table 31. Default Response Mode Values

Device Type	TRANRESP	NORESP
3270		X
FINANCE		X
LU 6	X	
NTO	X	
SLU 1	X	
SLU 2		X
SLU P		X

k is a MSGDEL option of SYSINFO, NONIOPCB, or NOTERM. The default is SYSINFO.

If OPTIONS=NOTERM is specified for a user descriptor, when that user descriptor is used for a sign on of a FINANCE, SLU P, or ISC terminal, SYSINFO issued instead of NOTERM.

The OPTION parameters are not positional. Refer to the MSGDEL and OPTIONS parameters of the TERMINAL macro for descriptions.

User Descriptor Examples

The following is an example of a user descriptor:

```
U SMITH ASOT=20 LTERM=(SECLT1) AUTLGN=SEC01LU2
```

Specifying the Number of Hash Table Slots

The number of hash table slots built by IMS for a statically defined resource (such as LTERMs, physical terminals, and users) is one-eighth the total number of the resource. To specify additional slots required for a resource created dynamically (conversations, LTERMs, physical terminals, and users), as a rule-of-thumb use one-eighth of the estimated total number of a dynamic resource. You might need to adjust this value. For information on setting the number of slots for specific hash tables, see the CHTS, LHTS, NHTS, and UHTS parameters on the IMS and DCC procedures.

Chapter 8. Accessing IMS Databases via the ODBA Interface

Overview

Open Database Access (ODBA) provides a callable interface that enables any OS/390 recoverable, resource-managed OS/390 address space to issue DL/I database calls to an IMS DB subsystem.

The interface provides OS/390 application programs (hereafter called ODBA applications) access to full-function DL/I databases and data entry databases (DEDBs). The ODBA application and IMS must coexist on the same OS/390 image.

The ODBA interface allows IMS DB and ODBA application programs to be developed, installed, and maintained independently. This independence provides failure isolation and resource recovery by using OS/390 Resource Recovery Services (RRS).

Related Reading: For a description of RRS/MVS and its uses, see Chapter 11, "Distributed Sync Point" in the *IMS/ESA Release Planning Guide*.

The ODBA interface resides in the OS/390 address space and is recognized by IMS as an application region (hereafter called the OS/390 application region).

The ODBA interface uses the Database Resource Adapter (DRA) to communicate with IMS DB.

Setting Up the DRA and the ODBA Interface

To set up ODBA and the DRA, you must follow these four steps:

1. Create the ODBA DRA startup table
2. the ODBA and DRA modules must reside in the STEPLIB or JOBLIN in the OS/390 application region
3. Link the ODBA application programs with DFSCDLI0
4. Set up security

Creating the ODBA DRA Start-up Table

You create the ODBA DRA startup table using the DFSPRP macro.

The startup table uses the DSNAME to dynamically allocate the data set that contains the rest of the ODBA interface routines. The DDNAME is generated to allow multiple connections to IMS from the same OS/390 application region. If you specify the DDNAME on the DFSPRP macro, it is ignored.

The default DSNAME is IMS.RESLIB. This is the default name established by the IMS generation process. This data set must be APF authorized.

Note: The IMS RESLIB library does not contain an DRA startup table. You must generate your own table by using the DFSPRP macro. Name the load module based on the following naming conventions:

- Characters 1-3 = DFS
- Characters 4-7 = specified 4-byte ID

The 4-byte ID should be the IMSID of the IMS system to which you will connect. However, this is not a requirement.

- Character 8 = 0

You must ensure that the DRA startup table module name is not the same as the name of an existing IMS module. To prevent accidental overlay, put the module in a load library that is accessible by the OS/390 application region.

If you use a different library for your own versions, make sure that the library is APF authorized. The DRA callable interface dynamically allocates the library by using the DD name and the data set name specified in the DRA startup table.

Loading and Running the ODBA and DRA Modules in the OS/390 Application Region

The following ODBA and DRA modules reside in the IMS RESLIB library. They must be in the STEPLIB or JOBLIB in the OS/390 application region.

- | | |
|-----------------|---|
| DFSCDLI0 | This module is linked or loaded by an application program. DFSCDLI0 also contains the ALIAS name AERTDLI. |
| DFSAERG0 | This module is loaded by DFSCDLI0. |
| DFSAERM0 | This module is attached by DFSAERG0 in the OS/390 application region. |
| DFSAERA0 | This module is attached by DFSAERM0 for initialization to the specified IMS DB subsystem. |

Linking Application Programs

You must link the ODBA application programs with DFSCDLI0 (AERTDLI). As an alternative, you can issue a load and branch command passing the AIB call list in Register 1.

Establishing and Defining Security

You must establish and define the connection security and PSB security to use for security control. For application programs that use the ODBA interface, use the APPLCTN macro to define the PSB names required by the ODBA applications.

APSB/Connection Security

Resource access from an unauthorized OS/390 application region connecting to the IMS DB environment is controlled by using the existing IMS DB ISIS execution parameter.

If you specify ISIS=1 or ISIS=2, both the OS/390 application region connection and PSB scheduling is checked. If you specify ISIS= 0, neither is done.

- Select ISIS=1

Before an OS/390 application region can connect to IMS DB, the USERID and Application Group Name (AGN) from the DRA Startup Table must be authorized to access the specified IMS DB.

RACF tables must be built that define valid user ID-AGN combinations. If the DRA Startup Table values do not correspond to an entry in RACF's tables, the OS/390 application region cannot connect to IMS DB.

Note: Important Connections to different IMS DB systems from same JOB can have different USERID/AGN security because different DRA Startup Table load modules can be loaded.

- Select ISIS=2

| Create a Resource Access Security exit routine, named DFSISIS0. This routine
| must determine whether the AGN passed to it is valid for the attempted
| connection.

PSB Security

| The current technique of searching the AGN table to determine if the PSB can be
| scheduled will continue to be used for PSB security. When ISIS=1, the ODBA
| interface will also use the SAF interface and do a RACHECK to validate the user
| associated with the caller's TCB is authorized to access the PSB.

ICMD Security

| For ICMD authorization, the ODBA application must set AIBRSNM1 field to the
| PSBNAME that was used on the previous APSB call. The PSBNAME is used by the
| existing ICMD support routines to issue the RACF call for security processing.

Chapter 9. Accessing IMS Databases with CICS

In this Chapter

- “Installing Data Sharing with CICS”
- “Installing CICS Intersystem Communication or Multiregion Operation with DL/I Buffer Handler Pool”
- “Installing CICS for Use with IMS Intersystem Communication” on page 416
- “Coding Considerations for PSBs” on page 416
- “Using Sequential Buffering” on page 417
- “CICS Connected to DL/I” on page 417
- “Accessing DBCTL Databases with CICS as a CCTL” on page 417

CICS can access DL/I databases via DBCTL. This means that DBCTL satisfies CICS DL/I requests by means of the CICS-DBCTL interface.

CICS/MVS provides support for IMS 3.1 local, remote and shared DB.

CICS/ESA provides support for IMS 3.1 DBCTL, local, remote, and shared DB.

Installing Data Sharing with CICS

CICS users can optionally use the IMS data sharing facility. The Internal Resource Lock Manager (IRLM) is mandatory for block-level data sharing, but not for database-level sharing.

Related Reading: For additional information, refer to *IMS/ESA Administration Guide: System*.

Installing CICS Intersystem Communication or Multiregion Operation with DL/I Buffer Handler Pool

You might want to install a system in which your DL/I databases are accessed by a CICS system in the same or separate processor, and are connected to the local system by a VTAM or MRO link.

You can also install a system in which your DL/I databases are not accessed by the local CICS system. If the local CICS system does not control any DL/I databases itself and IMS is not installed at the local site, a version of CICS can be generated to provide the minimal DL/I support required. This is done by specifying DLI=REMOTE when adding DL/I support to the supplied CICS system. Remote DL/I support only needs to be generated for CICS/MVS. It is pregenerated for CICS/ESA 3.1.

If DL/I is defined as remote, the IMS libraries are still required during stage 2 of the CICS system definition when generating EXEC DLI support and interregion or intersystem support. They are not required during stage 1. Therefore, these jobs might have to be done at a site where IMS is installed.

Installing CICS for Use with IMS Intersystem Communication

Related Reading: Information on defining CICS as an Intersystem Communication node and defining CICS tables to be compatible with IMS in an Intersystem Communication network is provided in *IMS/ESA Administration Guide: Transaction Manager*.

Coding Considerations for PSBs

This section provides some guidelines for coding PSBs.

Related Reading: For basic information on coding program specification blocks (PSBs), refer to *IMS/ESA Utilities Reference: System*.

PSBs for Online Transactions: A PSB is needed for each online program that accesses DL/I databases.

- The name of the PSB specified in the PSBNAME= keyword of the PSBGEN macro must be exactly the same as one of the entries in the PDIR.
- The name of the PSB must also be the same as the name specified in the scheduling call issued by an online transaction. If the online transaction does not specify any name in the scheduling call, the name of the PSB must be exactly the same as the name of the program associated with the transaction in the CICS program control table (PCT).

Example: Assume transaction X is associated with program Y. If program Y links or transfers control to program Z, which issues the scheduling call without specifying any PSB name, the default name for the PSB is Y, because it is program Y that is associated with transaction X in the PCT.

- The CMPAT=YES option on the PSBGEN statement can be omitted.
- There is no specific maximum size for an individual PSB that can be used in CICS/ESA. However, a limitation is set by the PSB pool size in DBCTL.

Additional Processing Intent Options (PROCOPT)

Two additional types of processing intent can be specified with the PROCOPT= keyword of the PCB or SENSEG statement. The two additional options are O and E.

PROCOPT=O specifies "read without integrity"; no dynamic enqueue is done by resource lock management for calls against the database. With the G intent option, you can specify GO, GOP, GON, or GOT. This option is only valid for the PCB statement. Read **Attention** note number 1 below for more information.

PROCOPT=E forces exclusive use of this database or segment by the online transaction. Other application programs scheduling a PSB referring to this database/segment wait during their scheduling process. No dynamic enqueue by resource lock management is done, but dynamic logging of database updates will be done. PROCOPT E can be specified with G, I, D, R, and A.

Attention:

1. The PROCOPT=O option affects integrity in reading and using uncommitted data. When you specify this option, IMS does not check the ownership of the segments returned. This means that the read-only user might have access to a segment that had been updated by another user. If the updating user then abends and is backed out, the read-only user would have seen a segment that

Coding Considerations for PSBs

no longer exists in the same form in the database. Consequently, if you specify this option, do not use the data that is read as a basis for updating records in any database.

2. An abend might occur with PROCOPT=GO if another program updates pointers when this program is following the pointers. Pointers are updated during insert, delete, and backout operations.
3. If PROCOPT=O in the PCB statement, the SENSEG statement must not specify a PROCOPT of I, R, D, H, or A.
4. If the O or E option is used, it must be coded immediately after the associated function code; for example GO, not OG.

Using Sequential Buffering

You can use sequential buffering with CICS. To do so, you need to do two things:

- Put an SBONLINE control statement in the //DFSVMxx file. SBONLINE allows sequential buffering to be used. For more information on SBONLINE, see “Specifying Sequential Buffering for an Online System” on page 350.
- Specify programs that are to use sequential buffering. You can do this by coding during PSBGEN an SB= keyword on the PCB macro.

Related Reading: For more information on the SB= keyword, see *IMS/ESA Utilities Reference: System*.

CICS Connected to DL/I

CICS can provide DL/I database support by using IMS. There are two ways in which you can use DL/I support with CICS:

- Through CICS remote DL/I support, also known as function shipping
- Through DBCTL

For CICS/ESA, remote DL/I support and DBCTL support are included in the pregenerated CICS.

For CICS/MVS, you must generate remote DL/I support.

Accessing DBCTL Databases with CICS as a CCTL

This section gives an overview of how to connect CICS to DBCTL.

Related Reading: See *IMS/ESA CICS-IMS Database Control Guide* for additional details.

DBCTL Tasks

This section describes DBCTL tasks associated with using DBCTL and CICS.

Defining CICS resources

If you use DBCTL exclusively, define the PSBs and DMBs using the APPLCTN and DATABASE macros.

If you want to function-ship requests to a remote CICS system, in which the database manager can be DBCTL or remote DL/I (function-shipping), you need to generate a PDIR only.

Accessing DBCTL Databases with CICS

Related Reading: Refer to *CICS System Definition Guide* and *CICS/ESA Resource Definition (Macro)* for information on defining PDIRs and DDIRs.

CICS routes DL/I requests to remote DL/I or DBCTL according to the PSB that is named. If the PSB does not appear in the CICS PDIR, and CICS is connected to DBCTL, CICS routes the request to DBCTL.

Initializing DBCTL

Use the procedure library member DBC to initialize the DBCTL subsystem. Also generated are procedures for DBRC and DL/I, which are used to initialize the DBRC and DL/I address spaces. The DBRC and DL/I procedures are started automatically by DBCTL during DBCTL address space initialization. All three procedures use positional parameters on the EXEC statement.

```
PARM='region type, parm1, parm2, parm3,...'
```

The region types specified are:

```
PARM='DBC' for DBCTL  
PARM='DRC' for DBRC  
PARM='DLS' for DLISAS
```

When all three address spaces have been initialized successfully, DBCTL issues:
DFS989I DBCTL READY

When the “READY” message is received, the IMS console operator enters a /START command. The commands options are:

- /NRESTART CHECKPOINT 0 for a coldstart with no previous shutdown
- /NRESTART for a warmstart
- /ERESTART for an emergency restart after a failure

CICS Tasks

This section describes CICS tasks associated with using DBCTL and CICS.

Defining DRA resources

The database resource adapter (DRA) is the interface between DBCTL and the transaction management subsystem. You need to put the DRA startup parameter table (DFSPZPxx) and the DRA startup router program (DFSPRRC0) in the CICS STEPLIB data set.

Example:

```
//STEPLIB DD DSN=CICS310.LOADLIB1,DISP=SHR  
// DD DSM=IMS.RESLIB,DISP=SHR
```

Related Reading: For more information on the DRA startup table see “Supporting CCTL Users with DBCTL Databases” on page 395, or “Using the Database Resource Adapter (DRA)” in the *IMS/ESA Application Programming: Database Manager*.

Connecting CICS to DBCTL

After CICS has been started, the CICS operator can issue a CONNECT command to DBCTL through the CDBC transaction. This transaction is also used for disconnecting from DBCTL. The CDBI transaction provides the status of the connection.

Related Reading: For more information on these transactions, see *IMS/ESA CICS-IMS Database Control Guide*.

Chapter 10. Accessing DB2 Databases with IMS

DB2 supports a relational database model. IMS message processing programs (MPPs), IMS Fast Path programs (IFPs), batch message programs (BMPs), IMS batch programs (DLI, DBB), and DBCTL BMPs can access both IMS hierarchic databases and DB2 relational databases. Complete documentation for DB2 is found in the library for the DB2 licensed program product.

This chapter describes installation tasks required for accessing DB2 databases.

- In your IMS subsystem, “What You Should Do for Your IMS Subsystem”:
 1. Define your DB2 subsystems to IMS:
 - a. Add a member to the IMS procedure library (IMS.PROCLIB) that contains the information about each DB2 subsystem with which IMS communicates.
 - b. On the execute procedures for the control region, message processing region, Fast Path region, or batch message processing region, specify the EXEC parameter SSM=, where SSM is the ID of the IMS.PROCLIB member which contains the DB2 subsystem definition.
 2. Define a language interface module, if you want to use one other than that supplied by IMS
 3. Specify on the IMS OPTIONS statement whether or not tracing of activity on the IMS-DB2 link will occur
- In your DB2 subsystems, “What You Should Do for Your DB2 Subsystem” on page 425:
 1. Provide the following two tables:
 - a. The external subsystem module table
 - b. The resource translation table (optional)
 - c. Ensure that the DB2 modules and tables used by IMS are in an appropriate APF-authorized library

What You Should Do for Your IMS Subsystem

This section describes tasks you must perform on your IMS system when you are connecting to a DB2 subsystem.

Defining Your DB2 Subsystems to IMS

Add a Member to the IMS Procedure Library

In order for IMS to initiate contact with a DB2 subsystem, it is necessary to add a member to the IMS procedure library, IMS.PROCLIB. This member contains an entry for each DB2 subsystem with which IMS is to communicate.

Using the IMS.PROCLIB data set allocated prior to stage 2 and the MVS utility IEBUPDTE create the PROCLIB members.

Related Reading: See *MVS/ESA Data Administration: Utilities* for information about the IEBUPDATE utility.

The first one to four characters of this PROCLIB member’s name must either match the IMS IMSID parameter specified on the IMSCTRL macro statement or be

Preparing the IMS Subsystem for DB2

overridden using the IMSID= EXEC parameter. The last four positions represent any unique installation-defined identifier. This identifier is appended to the IMS IMSID to construct the member name.

Each entry within the PROCLIB member consists of a blocked or unblocked 80-character record. There can be only one entry when using IMS batch. Information for each record begins in position 1 of the record. Fields within each entry are positional and are delimited by commas. Two consecutive commas denote the absence of an optional field. The fields in this record are:

SSN,LIT,ESMT,RTT,REO,CRC

Where:

SSN

Is a required one- to four-character MVS subsystem name of the DB2 subsystem that is to be attached. This name can be specified by your installation, and must have been identified to MVS during MVS system generation.

LIT

Is a required four-character alphanumeric field specifying the language interface token. The language interface provides routing information to IMS and also contains DB2-dependent information for the IMS applications. IMS provides a language interface module, named DFSLI000, which uses a token value of SYS1. To use this language interface, the token value SYS1 should be specified for the LIT parameter to relate this language interface to a DB2 subsystem. If you do not want to use this supplied language interface module, or want to use more than one (for example, to access test and production DB2 subsystems), you should assemble and link your user-defined language interface, with a token value other than SYS1, into an installation-selected library.

This token must then be specified on a record or, the standard language interface token, SYS1, or both. Each IMS control or dependent region can thereby access alternative DB2 subsystems via the corresponding language interface module. Each individual application program, however, can be link-edited to only one language interface module, and can thus access only one DB2 subsystem. Each application programmer must include the appropriate language interface module during the link-edit process for each application program that calls DB2 from an IMS MPP, IFP, or BMP.

For information on link-editing the language interface module into an appropriate IMS library or defining your own language interface module, see "Defining the Language Interface Module" on page 423.

ESMT

Is a required one- to eight-character alphanumeric parameter specifying the name of a table, called the external subsystem module table. It is defined in DB2 and specifies the DB2 modules IMS must load and the work areas that IMS must create.

DB2 must create the external subsystem module table and make the name of that table known to the IMS user who will define these PROCLIB member entries.

RTT

Is an optional one- to eight-character alphanumeric parameter specifying the name of the resource translation table. This table maps an application load

Preparing the IMS Subsystem for DB2

module name to a DB2 application plan, if these names are not the same. The resource translation table need only be used if IMS and DB2 use different names for the same resource.

DB2 must create the resource translation table and make the name of that table known to the IMS user who will define these PROCLIB member entries.

REO

Is an optional one-character region error option code. The option specified here determines the action IMS takes when an application program issues a request for DB2 services before connection to the DB2 subsystem is complete or if problems are encountered with the DB2 subsystem. IMS does not use these values, but passes them to DB2 for its use. The possible values for this parameter are:

- R** The application program receives a return code indicating that the request for DB2 services has failed. If no other value is specified for the region error option, R is the default. If an initial connection to DB2 cannot be established, no SQL return code is sent. The application terminates with an abend code U3047.
- Q** IMS abnormally terminates the application program with an abend code of U3051. IMS then places the transaction on a queue to be processed when the connection to DB2 is established. After a transaction is on this queue, it can be recovered across warmstarts and emergency restarts of IMS.
- A** IMS abnormally terminates the application program with an abend code of U3047 and discards the transaction input.

CRC

Is an optional one-character command recognition character. Any EBCDIC value except / is permitted. This parameter should be specified if you want to allow DB2 commands to be entered from IMS terminals, DBCTL operator, or automated operator interface (AOI) applications. Commands to be executed in DB2 are entered at an authorized IMS terminal or by an AOI application by first entering the /SSR command, followed by the command recognition character specified on this CRC parameter here, followed by the DB2 command. When IMS receives a command in this format, it routes that command to DB2 for processing.

Related Reading: For additional information, see the description of /SSR in *IMS/ESA Operator's Reference*.

The terminal from which the /SSR command is issued must be authorized to issue DB2 commands. If the terminal has a signed-on user, the user ID must be authorized to DB2. If no user exists, the logical terminal name must be authorized to DB2. IMS user IDs and logical terminal names are authorized to issue DB2 commands by adding them to the DB2 catalog, using the GRANT SQL statement.

Related Reading: Refer to *IBM DATABASE 2 Version 2 SQL Reference* for details on how to authorize IMS user IDs and logical terminal names to issue DB2 commands.

To authorize MVS consoles in a DBCTL environment to issue DB2 commands via the /SSR command, the system console name WTOR must be added to the DB2 catalog.

Preparing the IMS Subsystem for DB2

To use the defined PROCLIB member, you must:

- Insert a DD statement for IMS.PROCLIB with the ddname PROCLIB in the step execution JCL.
- Specify the correct PROCLIB member on the SSM= EXEC statement parameter.
- If an external subsystem connection is required to an IMS control region that is currently processing, dynamically specify an SSM member by issuing the /START SUBSYS SSM command. This allows a new subsystem connection to be started without stopping and then restarting the existing control region.

Specify the SSM EXEC Parameter: The SSM= EXEC parameter is required to enable your installation to indicate which PROCLIB member contains an entry for the DB2 subsystem to which you want to attach. SSM= can be specified on the EXEC procedures of that control region, an MPP, an IFP, or a BMP.

When SSM= is specified for the control region, any dependent region running under the control of the control region can attach to the DB2 subsystems named in the PROCLIB member pointed to by the SSM= parameter.

When SSM= is specified for a dependent region, it provides a “filter” that determines whether a given application program running in that dependent region can access DB2 resources.

For example, if a subsystem entry for a given DB2 subsystem exists in the named dependent region member, but not in the control region member, then a message will be sent to the MTO during dependent region initialization and a subsystem connection will not be made for this dependent region entry.

If SSM is specified on the control region but the installation wants a particular dependent region to be prohibited from attaching to any DB2 subsystem, a PROCLIB member with **no** entries must be created, and its ID must be specified in the dependent region. Application programs in this dependent region cannot attach to any DB2 subsystem.

When the PROCLIB member contains a null record, the following informational message is issued:

```
DFS3600I RC=18
```

Related Reading: Refer to *IMS/ESA Messages and Codes* for an explanation of this message.

If an invalid member name is specified on the SSM=EXEC parameter of the IMS control region JCL, no DB2 subsystem is attached, but IMS initialization continues. You can then enter the /START SUBSYS SSM command, specifying the correct subsystem name. If the /START SUBSYS SSM command is issued with an invalid subsystem member name, re-enter the command with a valid name.

If any record in the SSM PROCLIB member is invalid, a DFS3600 message is issued and processing continues. Valid records allow an external subsystem to be attached. Invalid records in the SSM PROCLIB member are ignored but can be corrected. Those external subsystems can be attached to the IMS control region using the /START SUBSYS command.

Defining the Language Interface Module

For an IMS application program to use the language interface module, the application program modules must be link-edited using the IMS reentrant DL/I language interface. The true attributes must be specified; for example, if a module is reusable only, don't specify "RENT".

The IMS/ESA DL/I language interface is not reentrant. Any IMS/ESA application programs that were designed to be reentrant or serially reusable can use DB2 resources only after being link-edited with the IMS language interface.

IMS and DB2 share a common alias name, DSNHLI, for the language interface module. Therefore, use caution when concatenating your libraries to ensure the correct order.

- If you are running an IMS dependent region with an application program using the COBOL dynamic option, the IMS library must be concatenated first.
- If you are running an application program strictly under DB2, be sure the DB2 library is concatenated first.

IMS provides a language interface module, DFSLI000. You can optionally define your own language interface modules with unique LITs. To assist you in doing this, IMS provides the DFSLI macro, the expansion of which generates the required code. The format of this macro is as follows, where user-supplied values can be substituted for the x's.

Example:

```
DFSLIxxx DFSLI TYPE=V2DB,LIT=xxxx
        END
```

where:

DFSLIxxx

Specifies the CSECT name for the module. It is recommended that the CSECT name match the link-edit name of the language interface module. For the IMS-supplied language interface module, the value for this parameter is DFSLI000.

TYPE

Specifies the language interface type to IMS. This example is for DB2 V2.

LIT

Specifies the language interface token that relates this language interface to a DB2 subsystem by means of their respective specifications in a PROCLIB member entry as the LIT and SSN parameters. The IMS-supplied language interface module DFSLI000 uses SYS1 as the LIT value.

When IMS issues a DB2 request, IMS knows the target subsystem by the LIT used in the request.

Example: Consider the case of an IMS dependent region accessing two DB2 subsystems, DSN1 and DSN2.

- You already have a default language interface module DFSLI000, as specified by the default LIT, SYS1. You now generate a second language interface DFSLI001 with a LIT of SYS2.
- Define two entries in the PROCLIB SSM member. The first entry points to DSN1 with LIT=SYS1. The second entry points to DSN2 with LIT=SYS2.

Preparing the IMS Subsystem for DB2

- Link-edit IMS application programs accessing DSN1 with the IMS-provided language interface module DFSLI000.
- Link-edit IMS application programs accessing DSN2 with the language interface module that you have generated, DFSLI001.

Although a region can communicate with two or more DB2 subsystems, an IMS application can access only one— the DB2 subsystem referred to in the link-edited language interface module. You can, however, alter the SSM member to route application requests to a DB2 subsystem other than the subsystem defined in the language interface module.

Specifying the Desired Trace Options

If you want to trace the connection and disconnection of IMS to DB2, the parameter SUBS= on the OPTIONS statement must be set to ON. Specifying OUT on this parameter causes trace information to be written to the log. If this parameter is omitted, traces can be turned on and off by means of the /TRACE command.

Specifying DB2 for IMS Batch Regions

When using DB2 Version 2 Release 2 and later versions, the SSM= parameter can be used in IMS DL/I and DBB batch regions to specify the DB2 connection parameters instead of the DDITV02 DD statement. Refer to the DB2 library for more information on using DB2 in IMS batch regions.

The subsystem member in PROCLIB is the same format as for online; however, only a single DB2 subsystem can be specified. The LIT token used is SYS1, and the following additional parameters are passed to DB2 by IMS in batch regions:

- Connection Name. IMS uses the job name for the batch region as the connection name to provide uniqueness within the MVS/ESA system for DB2.
- Plan. IMS uses the application program name as the plan name. The resource translation table (RTT) can be used to translate the program name to the desired plan name if the plan has another name.
- Program Name. The application program name can be specified in one of the following ways:
 - DSNMTV01 can be specified as the application program name for the batch region. When this method is used, control is given to DSNMTV01. When the DB2 environment is established, control is passed to the application program.
 - The application program name can be specified for the batch region. When this method is used, the SSM= parameter must be specified on the batch region parameters. Control is given to DSNMTV01 directly to establish the DB2 environment. When the DB2 environment is established, control is passed to the application program specified in the batch region.

The subsystem member allows a batch region to use DB2 without specifying the DB2 parameters and application program in the batch region JCL. However, the DDITV02 DD statement can still be used. In fact, the SSM= parameter is overridden if both the DDITV02 DD statement is used and DSNMTV01 is the batch application program name.

The DDOTV02 DD statement must be used for messages and diagnostic information regardless of how the DB2 connection parameters are specified.

What You Should Do for Your DB2 Subsystem

This sections describes tasks you must perform to prepare your DB2 subsystem for use with IMS.

Providing Appropriate Tables in DB2

It is the responsibility of the DB2 user to provide the external subsystem module table and, optionally, the resource translation table in the DB2 subsystem. It is also the DB2 user's responsibility to make the names of these tables known to the person who is responsible for defining the IMS.PROCLIB member entries. The person defining the PROCLIB entries includes the names of these tables as values for the parameters ESMT and RTT, respectively.

Placing DB2 Modules and Tables in the Appropriate Library

IMS loads the modules and tables specified in the external subsystem module table from DB2 libraries when the SSM EXEC parameter is specified in an IMS region. To ensure that this process is successful, you must make the DB2 libraries, which must be APF-authorized, available to IMS. For this purpose, you can either add the DB2 libraries to the JOBLIB/STEPLIB/LINKLIST concatenation or create a DFSESL DD statement for this purpose. (IMS does not automatically generate a DFSESL DD statement.)

If the JOBLIB/STEPLIB/LINKLIST concatenation is not authorized, you must use the DFSESL DD statement. For online IMS regions, the subsystem library or libraries must be concatenated after the library containing the IMS modules (usually IMS.RESLIB). When multiple subsystems are connected, additional subsystem data sets can be concatenated.

Example:

```
//DFSESL DD DISP=SHR,DSN=IMS.RESLIB
//          DD DISP=SHR,DSN=DSNxxx.DSNLOAD
//          DD DISP=SHR,DSN=DSNyyy.DSNLOAD
```

IMS.RESLIB is first, followed by the subsystem libraries.

For IMS batch regions, the DFSESL DD statement is not used. The DB2 libraries must be in both the JOBLIB/STEPLIB/LINKLIST concatenation and the DFSRESLB DD statement for authorized modules.

If you use the COBOL II dynamic option, the DB2 libraries must be added to the JOBLIB or STEPLIB regardless of authorization. The IMS.RESLIB DD statement must precede the reference to DSNxxx.DSNLOAD in the JOBLIB or STEPLIB. If the JOBLIB or STEPLIB is not authorized, you must also use a DFSESL DD statement.

Appendix A. IMS System Definition (Step 3) Examples

The IMS system definitions in this appendix are representative samples of the stage 1 input to the assembler during an IMS system definition. The following sample definitions are presented:

- “IMS DB/DC Environment”
- “Data Communication Macro Statements” on page 434
- “IMS Multiple Systems Coupling” on page 450
- “Data-Sharing System Configuration” on page 455
- “IMS DBCTL Environment” on page 458

An example of IMS system definition stage 1 input, which defines all supported terminal types, is provided as a COPY member of IMS.GENLIBB (VSINPUT). To obtain an up-to-date example of input to, and output from, stage 1 of IMS system definition, use the following JCL and specify MACLIB=ALL on the IMSGEN macro statement:

```
// JOB
// EXEC  ASMFC,REGION=2048K
//ASM.SYSLIB DD DSNAME=IMS.GENLIB,DISP=SHR
//          DD DSNAME=IMS.GENLIBA,DISP=SHR
//          DD DSNAME=IMS.GENLIBB,DISP=SHR
//ASM.SYSIN DD DSNAME=IMS.GENLIBB(VSINPUT),
//          DISP=SHR
```

The IMS system definition macro statements shown below are the only ones necessary for the system definition for an MVS SP3.1 database system.

Characteristics to Define	Macro Statements Prepared	Column 72
IMS Configuration	IMSCTRL SYSTEM=(VS/2,(BATCH,DB/DC),4.3) IMSGEN ASM=HLASM,ASMPRT=ON, LKPRT=(XREF,LIST), MACLIB=ALL,JOBCTL=(4,C,T), PRTY=6,PROCLIB=YES	X X X

IMS DB/DC Environment

A hypothetical IMS DB/DC environment is the basis for this example of an IMS DB/DC system definition. An overview of the IMS system is followed by system details and macros.

In this example, the system has the following characteristics:

- 46 application programs
- 67 transaction codes against those application programs
- 16 databases

Data Communication Characteristics

The data communication characteristics of the hypothetical environment include:

- Four local SYSOUT line groups
- One 2740 nonswitched line group with two lines
 - The first line has two 2740 Model 1 terminals

- The second line has two 2740 Model 2 terminals
- One 2740 non-station control line group containing one line and one terminal
- One 2741 nonswitched line group containing one line and one terminal
- One 3270 local line group containing one line and six terminals
- One 3270 local line group containing one line and two 3286 printer terminals
- One 3270 remote line group containing one line, one 3275 terminal, and one 3271 control unit supporting five terminals
- One System/3 line group containing one line, with one station supporting four terminals
- One System/7 start/stop line group containing one line, with one station supporting four terminals
- One System/7 BSC line group containing one line, with one station supporting one terminal
- Two 2741 switched line groups
- One 3275 switched line group with two lines
- Six 3270 local VTAM terminals
- Six 3270 remote VTAM terminals
- Five 3600 (or FINANCE) VTAM terminals
- Three SLU 1 VTAM terminals
- Two NTO terminals
- Two SLU 2 VTAM terminals
- Two SLU P VTAM terminals
- Two LU 6.1 VTAM terminals

System Configuration Macro Statements

The system configuration macros below select IMS system functions for MVS, allowing 5 regions to operate simultaneously. They specify 16 transaction code classes, and the number of concurrently operating subtasks optimized to equal the number of specified communication lines divided by two.

Characteristics to Define	Macro Statements Prepared	Column
IMS Configuration	IMSCTRL SYSTEM=(VS/2,(ALL,DB/DC),4.3),	X
	MAXREGN=(5,52K,A,3),	X
	MCS=13,DESC=2,MAXCLAS=16,	X
	IMSID=IMSA	
	IMSCTF SVCNO=(,254,255),	X
	APNDG=(,ZZ),	X
	RDS=(3330,2048),	X
	CPLOG=2500	
System Buffers and Data Sets	BUFPOOLS PSB=8000,PSBW=6000,	X
	DMB=11000,FRE=28,	X
	MSGQUEUE DSETS=3330,	X
	BUFFERS=(20,1152),	X
	SHUTDWN=150	
Security Facilities	SECURITY TRANCMD=NO,TERMNL=NO,	X
	PASSWD=NO,	X
	TYPE=(NORACTRM,NOSIGNEX,NOTRANEX,NOAGN),	X
	SECLVL=(NOTRAN,NOSIGN)	

Database and Application Macro Statements

The system has the following DL/I databases.

Characteristics to Define	Macro Statements Prepared	Column 72
DL/I Databases		
DH41SK01, DH41SK02, DH41TS01	DATABASE DBD=(DH41SK01,DH41SK02,DH41TS01)	
DI31LM01	DATABASE DBD=DI31LM01	
DI21PART, DH41SK03, DX41SK01	DATABASE RESIDENT, DBD=(DI21PART,DH41SK03,DX41SK01)	X
DI41SK01, DI41SK02	DATABASE DBD=(DI41SK01,DI41SK02)	
DD41SK01, DD41TS01, DX41TS01	DATABASE DBD=(DD41SK01,DD41TS01,DX41TS01)	
LTERMINL, LTINDXDB	DATABASE DBD=(LTERMINL,LTINDXDB)	
TESTABLE, TTINDXDB	DATABASE DBD=(TESTABLE,TTINDXDB)	

Application Programs

Each DL/I application program is shown with associated transaction codes and macros.

Characteristics to Define	Macro Statements Prepared	Column 72
Programs and Transaction Codes		
CONTEST	APPLCTN PSB=CONTEST	
CONV1	TRANSACT CODE=CONV1,PRTY=(8,8), PROCLIM=(3,60),SPA=80, MODE=SNGL	X X
CONV2	TRANSACT CODE=CONV2,PRTY=(8,8), PROCLIM=(3,60), SPA=80,MODE=SNGL	X X
CONV3	TRANSACT CODE=CONV3,PRTY=(8,8), SPA=80, MODE=SNGL	X X
CONV4	TRANSACT CODE=CONV4,PRTY=(8,8), SPA=80, MODE=SNGL	X X
ENQ0SK41	APPLCTN PSB=ENQ0SK41	
ENQ	TRANSACT CODE=ENQ,PRTY=(8,8), MODE=SNGL	X
HDTASK01	APPLCTN PSB=HDTASK01	
SKD1	TRANSACT CODE=SKD1,PRTY=(8,8)	
HHBLSK41 (Batch)	APPLCTN PSB=HHBLSK41,PGMTYPE=BATCH	
HHBLSK42 (Batch)	APPLCTN PSB=HHBLSK42,PGMTYPE=BATCH	
HHBLSK43 (Batch)	APPLCTN PSB=HHBLSK43,PGMTYPE=BATCH	
HHTASK41	APPLCTN PSB=HHTASK41, PGMTYPE=(,2), SCHDTYP=PARALLEL	X X
SKH1	TRANSACT CODE=SKH1,PRTY=(8,8), PARLIM=1	X
HHTASK42	APPLCTN PSB=HHTASK42,PGMTYPE=(,3)	
SKH2	TRANSACT CODE=SKH2,PRTY=(8,8)	
HHTASK43	APPLCTN PSB=HHTASK43,PGMTYPE=(,4)	
SKH3	TRANSACT CODE=SKH3,PRTY(8,8)	
HIBASK41 (Batch)	APPLCTN PSB=HIBASK41,PGMTYPE=BATCH	
HITASK41	APPLCTN PSB=HITASK41,PGMTYPE=(,2)	
SKI1	TRANSACT CODE=SKI1,PRTY=(8,8)	
HITASK42	APPLCTN PSB=HITASK42,PGMTYPE=(,3)	
SKI2	TRANSACT CODE=SKI2,PRTY(8,8)	

Characteristics to Define	Macro Statements Prepared	Column 72
Programs and Transaction Codes		
HSBASK41 (Batch)	APPLCTN PSB=HSBASK41,PGMTYPE=BATCH	
HIMAJC01 TPPL1	APPLCTN PSB=HIMAJC01 TRANSACT CODE=TPPL1,PRTY=(8,8)	
HIMAJC03 TUBE	APPLCTN PSB=HIMAJC03 TRANSACT CODE=TUBE,PRTY=(8,8), SPA=100,MODE=SNGL	X
TUBFD	TRANSACT CODE=TUBFD,PRTY=(8,8), SPA=100, MODE=SNGL	X X
TUBFC	TRANSACT CODE=TUBFC,PRTY=(8,8), SPA=100, MODE=SNGL	X X
TUBMA	TRANSACT CODE=TUBMA,PRTY=(8,8), SPA=20000,MODE=SN GL	X
HIMALM01	APPLCTN PSB=HIMALM01, SCHDTYP=PARALLEL	X
DLI	TRANSACT CODE=DLI,PRTY=(5,10,5), PROCLIM=(10,10), MSGTYPE=(SNGLSEG,RESPONSE)	X X
ICS	TRANSACT CODE=ICS,PRTY=(5,12,5), PROCLIM=(10,100), MODE=SNGL	X X
DLN	TRANSACT CODE=DLN,PRTY=(0,8,3), PROCLIM=(10,100)	X
IMS	TRANSACT CODE=IMS,PRTY=(5,12,5), PROCLIM=(1,100), MSGTYPE=SNGLSEG,PARLIM=1	X X
CONALTR CONAL	APPLCTN PSB=CONALTR TRANSACT CODE=CONAL,MODE=SNGL, PRTY=(8,8),SPA=100	X
CONALF	TRANSACT CODE=CONALF,MODE=SNGL, PRTY=(8,8), SPA=100	X X
LKMDFS00 STL	APPLCTN PSB=LKMDFS00 TRANSACT CODE=STL,PRTY=(5,12,5), PROCLIM=(8,100),INQ=YES	X
LKMDFS10 LKM	APPLCTN PSB=LKMDFS10,PGMTYPE=BATCH TRANSACT CODE=LKM,PRTY=(0,0), MSGTYPE=SNGLSEG	X
MR1 MR1	APPLCTN PSB=MR1 TRANSACT CODE=MR1,PRTY=(8,8)	
MR2 MR2	APPLCTN PSB=MR2 TRANSACT CODE=MR2,PRTY=(8,8)	
MR3 MR3	APPLCTN PSB=MR3 TRANSACT CODE=MR3,PRTY=(8,8)	
MR4 MR4	APPLCTN PSB=MR4 TRANSACT CODE=MR4,PRTY=(8,8)	
MR5 MR5	APPLCTN PSB=MR5 TRANSACT CODE=MR5,PRTY=(8,8)	

Characteristics to Define	Macro Statements Prepared	Column 72
Programs and Transaction Codes		
MR6 MR6	APPLCTN PSB=MR6 TRANSACT CODE=MR6,PRTY=(8,8)	
TESTIMSD TSTCONV	APPLCTN PSB=TESTIMSD TRANSACT CODE=TSTCONV,PRTY=(8,8), SPA=100,MODE=SN GL	X
TSTNORM	TRANSACT CODE=TSTNORM,PRTY=(8,8)	
REQIMSD TREQ	APPLCTN PSB=REQIMSD TRANSACT CODE=TREQ,PRTY=(8,8), SPA=100,MODE=SNGL	X
LOADLT (Batch)	APPLCTN PSB=LOADLT,PGMTYPE=BATCH	
LOADTT (Batch)	APPLCTN PSB=LOADTT,PGMTYPE=BATCH	
STLECHO ECHO	APPLCTN PSB=STLECHO TRANSACT CODE=ECHO,PRTY=(5,10,5), PROCLIM=(10,10), MSGTYPE=SNGLSEG	X X
ECHO	TRANSACT CODE=ECHO,PRTY=(5,10,5), PROCLIM=(10,10)	X
MR8 MR8	APPLCTN PSB=MR8 TRANSACT CODE=MR8,PRTY=(8,8)	

Characteristics to Define	Macro Statements Prepared	Column 72
--------------------------------------	----------------------------------	----------------------

Programs and
Transaction
Codes

RESIDENT PSBS

RESPSB1 RESPSB	APPLCTN PSB=RESPSB1,RESIDENT TRANSACT CODE=RESPSB,PTY=(8,8)	
RESBAL01 BAL01 BAL02 BAL03	APPLCTN PSB=RESBAL01, SCHDTYP=PARALLEL,RESIDENT TRANSACT CODE=(BAL01,BAL02,BAL03), PTY=(8,8)	X X
RESBAL02 BALTRAN	APPLCTN PSB=RESBAL02, SCHDTYP=PARALLEL TRANSACT CODE=BALTRAN,PTY=(8,8), PARLIM=1	X X
PSBEXINT TRANEXNT	APPLCTN PSB=PSBEXINT TRANSACT CODE=TRANEXNT,PTY=(8,8)	
INTCON ANY HI EQHI NXCLASS	APPLCTN PSB=INTCON TRANSACT CODE=ANY,PTY=(8,8),SCHD=3 TRANSACT CODE=HI,PTY=(8,8),SCHD=2 TRANSACT CODE=EQHI,PTY=(8,8), SCHD=1 TRANSACT CODE=NXCLASS,PTY=(8,8), SCHD=4	X X
CUTOFF SIX SEVEN EIGHT NINE	APPLCTN PSB=CUTOFF TRANSACT CODE=SIX,PTY=(6,6) TRANSACT CODE=SEVEN,PTY=(7,7) TRANSACT CODE=EIGHT,PTY=(8,8) TRANSACT CODE=NINE,PTY=(9,9)	
NXTCLASS ONE	APPLCTN PSB=NXTCLASS,PGMTYPE=(,2) TRANSACT CODE=ONE,PTY=(1,1)	

The system includes one 3600 application program.

DC Macro Statements

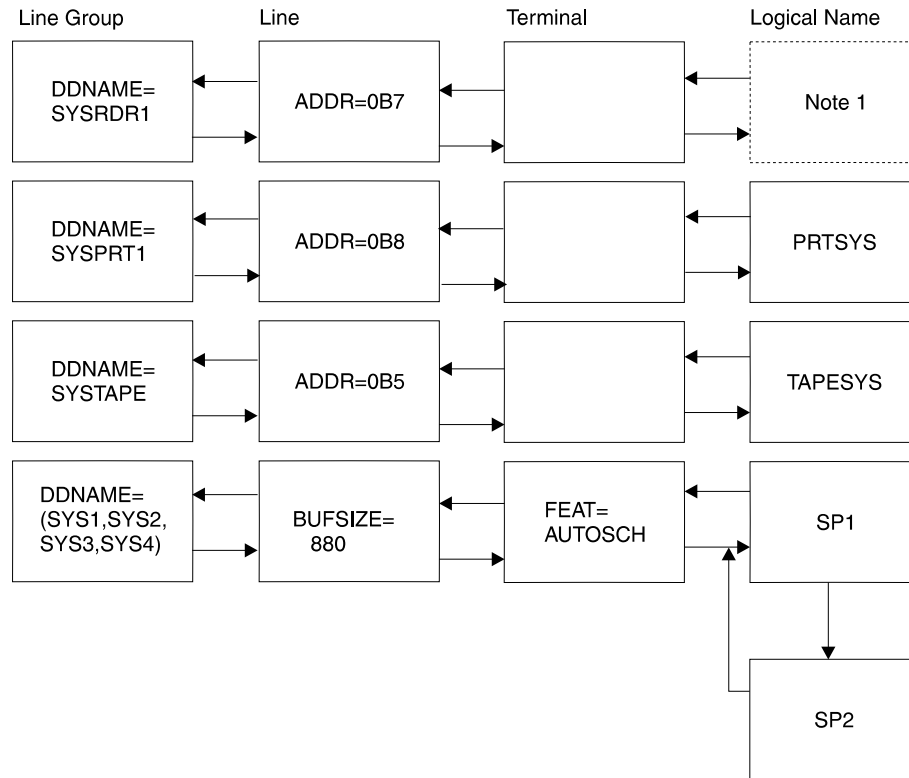
Characteristics to Define	Macro Statements Prepared	Column 72
Programs and Transaction Codes		
TEST3600	APPLCTN DOPT,PSB=TEST3600, PGMTYPE=TP	X
CIFINQ	TRANSACT CODE=CIFINQ,PRTY=(1,2,2), MSGTYPE=SNGLSEG, PROCLIM=(,30),INQUIRY=YES, SEGN0=256,SEGSIZE=4096	X X X
SAVDBT	TRANSACT CODE=(SAVDBT,SAVDPT), PRTY=(1,2,2), MSGTYPE=SNGLSEG, PROCLIM=(,30),INQUIRY=NO, SEGN0=10,SEGSIZE=80	X X X X
BRNCHTOT	TRANSACT CODE=BRNCHTOT, PRTY=(1,2,2),MSGTYPE=SNGLSEG, PROCLIM=(,30),INQUIRY=YES, SEGN0=256,SEGSIZE=132	X X X
PGM3741 TX3741	APPLCTN PSB=PGM3741,PGMTYPE=TP TRANSACT CODE=TX3741	
DFSSAM02	APPLCTN PSB=DFSSAM02, PGMTYPE=(TP,,4)	X
PART	TRANSACT CODE=PART,PRTY=(7,10,2), INQ=(YES,NORECOV)	X
DFSSAM03 DSPINV	APPLCTN PSB=DFSSAM03 TRANSACT CODE=DSPINV,PRTY=(7,10,2), INQ=YES,MSGTYPE=(,4)	X
DFSSAM04 ADDPART	APPLCTN PSB=DFSSAM04,PGMTYPE=(,4) TRANSACT CODE=(ADDPART,DLETPART), PRTY=(7,10,2)	X
DLETPART ADDINV	TRANSACT CODE=(ADDINV,DLETINV), PRTY=(7,10,2),MSGTYPE=(,5)	X
DLETINV		
DFSSAM05 CLOSE	APPLCTN PSB=DFSSAM05,PGMTYPE=(,3) TRANSACT CODE=CLOSE,PRTY=(7,10,2)	
DFSSAM06 DISBURSE	APPLCTN PSB=DFSSAM06 TRANSACT CODE=DISBURSE, PRTY=(7,10,2),MSGTYPE=(,2)	X
DFSSAM07 DSPALLI	APPLCTN PSB=DFSSAM07,PGMTYPE=(,2) TRANSACT CODE=DSPALLI, PRTY=(7,10,2),INQ=YES	X

Data Communication Macro Statements

BTAM, BSAM, GAM, and ARAM Macro Sets

The following macros describe the BTAM facilities. A graphic representation is presented, followed by examples of the required macros statements.

Local SYSOUT Line Group



Note:

1. Refers to an LTERM named using the TERMINAL and NAME (LTERM name) system definition statement.

Macro Statement Set Prepare

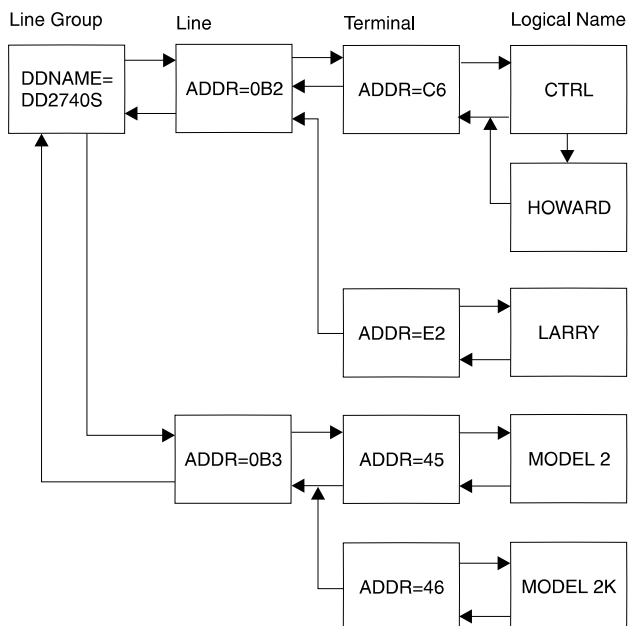
d

```

LINEGRP DDNAME=SYSRDR1,UNITYTYPE=READER
LINE ADDR=0B7
TERMINAL LTERM=T2741
LINEGRP DDNAME=SYSPRT1,UNITYTYPE=PRINTER
LINE ADDR=0B8
TERMINAL
NAME PRTSYS
LINEGRP DDNAME=SYSTAPE,UNITYTYPE=TAPE
LINE ADDR=0B5,BUFSIZE=220
TERMINAL
NAME TAPESYS
LINEGRP UNITYTYPE=SPOOL,DDNAME=(SYS1,SYS2,SYS3,SYS4)
LINE BUFSIZE=880
TERMINAL FEAT=AUTOSCH
NAME SP1,SP2
    
```

DC Macro Statements

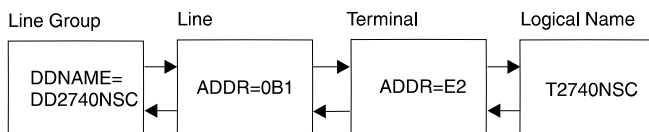
2740 Nonswitched Line Group



Macro Statement Set Prepared

```
LINEGRP DDNAME=DD2740S,UNITYPE=2740
LINE ADDR=0B2
  TERMINAL ADDR=C6,OPTIONS=MFS
  NAME CTRL,HOWARD
  TERMINAL ADDR=E2,OPTIONS=MFS
  NAME LARRY
LINE ADDR=0B3,MODEL=(2,440)
  TERMINAL ADDR=45,BUFSIZE=248,OPTIONS=(MFS)
  NAME MODEL2
  TERMINAL ADDR=46,OPTIONS=MFS
  NAME MODEL2K
```

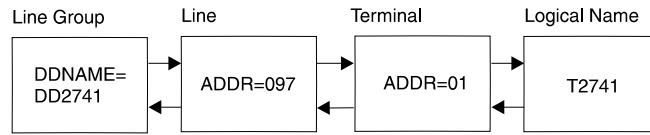
2740 Non-Station Control Line Group



Macro Statement Set Prepared

```
LINEGRP DDNAME=D2740NSC,UNITYPE=(2740,NOSTACTL)
LINE ADDR=0B1
  TERMINAL ADDR=E2,OPTIONS=MFS
  NAME T2740NSC
```

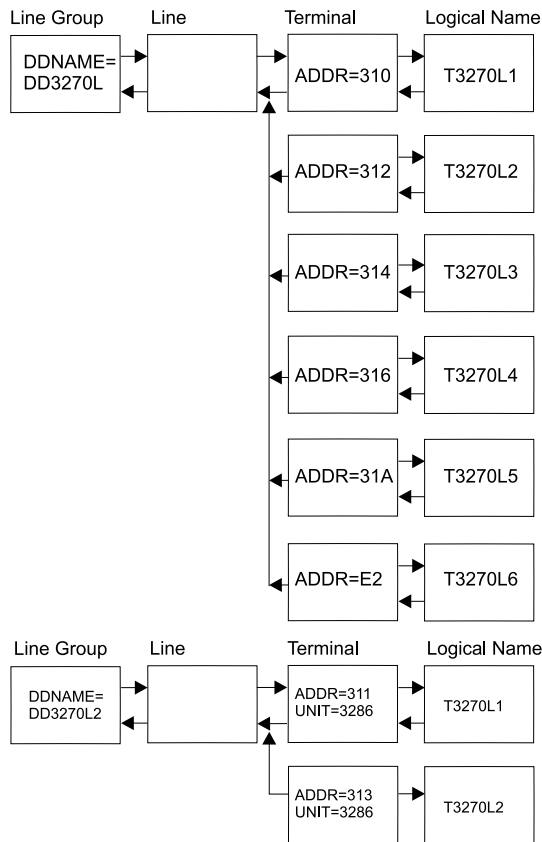
2741 Nonswitched Line Group



Macro Statement Set Prepared

```
LINEGRP DDNAME=DD2741,UNITYPE=2741,CODE=CORRS
LINE ADDR=097
  TERMINAL ADDR=01,FEAT=NOINTRPT,OPTIONS=MFS
  NAME T2741
```

3270 Local Line Groups

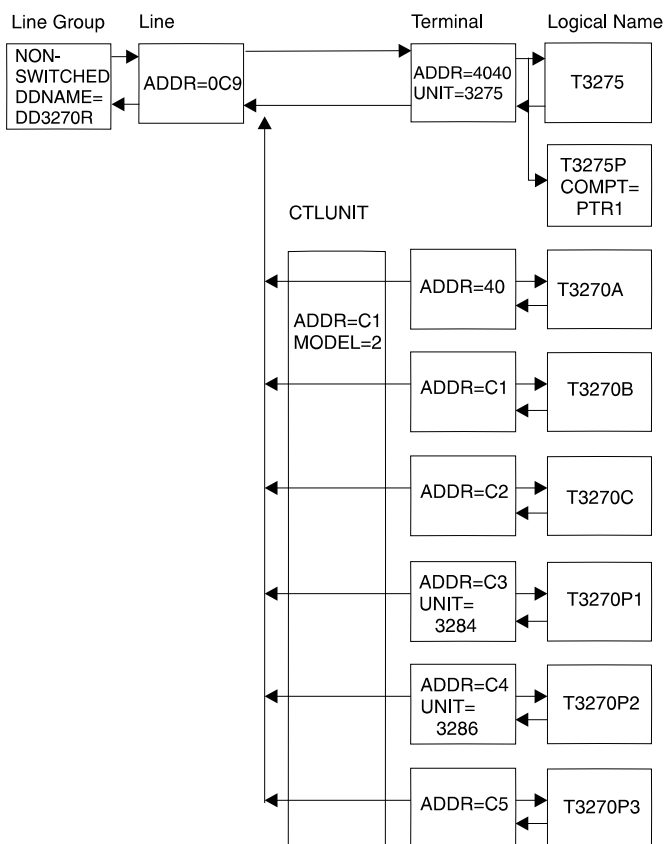


DC Macro Statements

```

Macro Statement Set Prepared
Column
72
LINEGRP DDNAME=DD3270L,UNITYPE=(3270,LOCAL)
LINE BUFSIZE=300
  TERMINAL ADDR=310,MODEL=2
  NAME T3270L1
  TERMINAL ADDR=312,FEAT=(PFK,CARD,PEN),
  OPTIONS=FORCRESP,MODEL=1
  NAME T3270L2
  TERMINAL ADDR=314,TYPE=3270-A1,SIZE=(12,80)
  NAME T3270L3
  TERMINAL ADDR=316,TYPE=3270-A2,SIZE=(24,80)
  NAME T3270L4
  TERMINAL ADDR=318,TYPE=3270-A5,SIZE=(12,40)
  NAME T3270L5
  TERMINAL ADDR=31A,TYPE=3270-A6,SIZE=(6,40)
  NAME T3270L6
LINEGRP DDNAME=DD3270L2,UNITYPE=(3270,LOCAL)
LINE
  TERMINAL ADDR=311,MODEL=2,UNIT=3286
  NAME T3270P3
  TERMINAL ADDR=313,MODEL=1,UNIT=3286
  NAME T3270P4
  
```

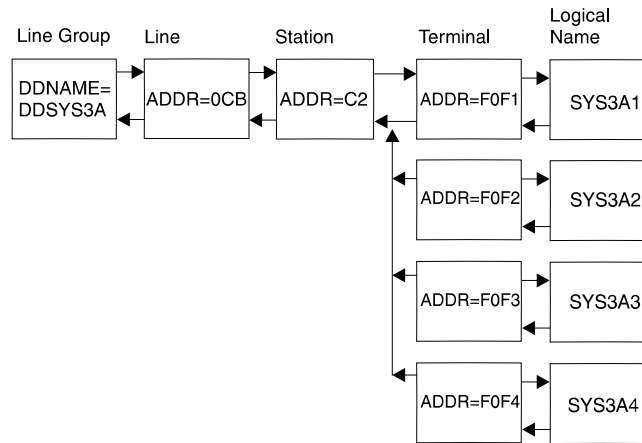
3270 Remote Line Group



```

Macro Statement Set Prepared
Column
72
LINEGRP DDNAME=DD3270R,UNITYPE=3270,CODE=EBCDIC
LINE ADDR=0C9
  TERMINAL ADDR=4040,UNIT=3275,COMPT=PTR1,
    FEAT=PFK,OPTIONS=COPY
  NAME T3275
  NAME T3275P,COMPT=PTR1
CTLUNIT ADDR=C1,MODEL=2
  TERMINAL ADDR=40,FEAT=(PFK,CARD,PEN),
    MODEL=2,OPTIONS=COPY
  NAME T3270A
  TERMINAL ADDR=C1,FEAT=(PFK,CARD,PEN),
    MODEL=2,OPTIONS=COPY
  NAME T3270B
  TERMINAL ADDR=C2,FEAT=(PFK,CARD,PEN),
    MODEL=1,OPTIONS=COPY
  NAME T3270C
  TERMINAL ADDR=C3,UNIT=3284,MODEL=2,PTRSIZE=132
  NAME T3270P1
  TERMINAL ADDR=C4,UNIT=3286,MODEL=2,PTRSIZE=132
  NAME T3270P2
  TERMINAL ADDR=C5,FEAT=F1,TYPE=3270-A2,
    SIZE=(24,80)
  NAME T3270D
  
```

System/3 Line Group

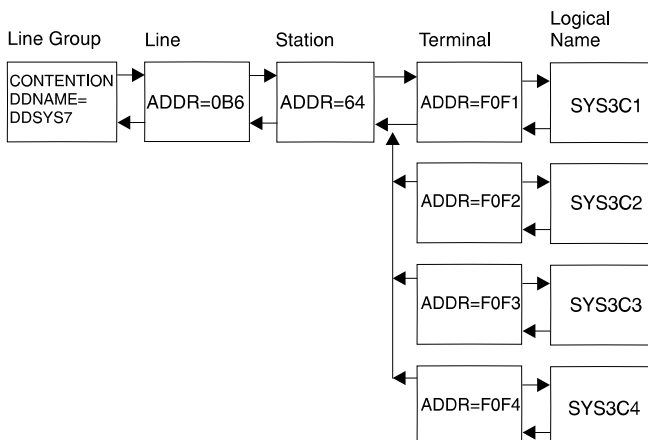


```

Macro Statement Set Prepared
LINEGRP DDNAME=DDSYS3A,UNITYPE=S/3
LINE ADDR=0CB,BUFSIZE=(256,256)
STATION ADDR=C2,TRANSLM=12,TYPE=ASK
  TERMINAL ADDR=F0F1
  NAME SYS3A1
  TERMINAL ADDR=F0F2
  NAME SYS3A2
  TERMINAL ADDR=F0F3
  NAME SYS3A3
  TERMINAL ADDR=F0F4
  NAME SYS3A4
  
```

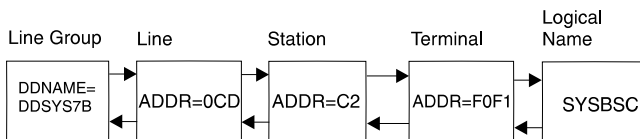
DC Macro Statements

System/7 Line Group



Macro Statement Set Prepared

```
LINEGRP DDNAME=DDSYS7,UNITYTYPE=S/7
LINE ADDR=0B6,BUFSIZE=(202,82)
STATION ALIGN=2,TRANSLM=12,ADDR=64,TYPE=ASK
TERMINAL ADDR=F0F1
NAME SYS7C1
TERMINAL ADDR=F0F2
NAME SYS7C2
TERMINAL ADDR=F0F3
NAME SYS7C3
TERMINAL ADDR=F0F4
NAME SYS7C4
```



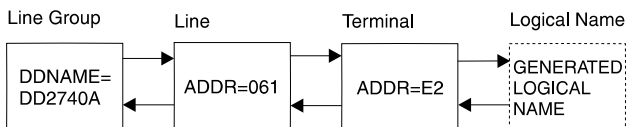
Macro Statement Set Prepared

```
LINEGRP DDNAME=DDSYS7B,UNITYTYPE=(S/7,,BSC)
LINE ADDR=0CD,BUFSIZE=(256,256)
STATION ADDR=C2,TRANSLM=12,TYPE=ASK
TERMINAL ADDR=F0F1
NAME SYS7BSC
```

Switched Communication Device Macro Sets

The following sets of macro statements are required to describe the switched communication devices in the hypothetical system. A graphic representation of each of these is presented, along with the macro statements required to define it.

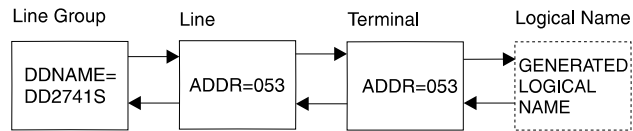
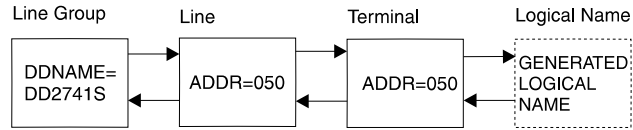
2740 Switched Line Group



Macro Statement Set Prepared

```
LINEGRP DDNAME=DD2740A,UNITYPE=2740,FEAT=SWITCHED
LINE ADDR=061
  TERMINAL ADDR=E2,OPTIONS=MFS
```

2741 Switched Line Group



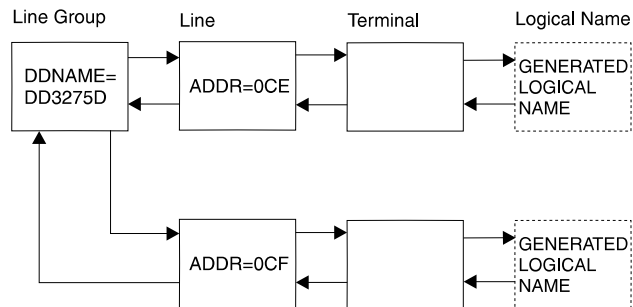
Macro Statement Set Prepared

Column

```
LINEGRP DDNAME=DD2741S,UNITYPE=2741,          72
        CODE=(CORRS,EBCDIC),FEAT=SWITCHED      X
LINE ADDR=050
  TERMINAL ADDR=50,OPTIONS=MFS

LINEGRP DDNAME=DD2741S2,UNITYPE=2741,         X
        CODE=(CORRS,BCD),FEAT=SWITCHED
LINE ADDR=053
  TERMINAL ADDR=53,OPTIONS=MFS
```

3275 Switched Line Group



DC Macro Statements

Macro Statement Set Prepared

```

LINEGRP UNITYPE=3275,DDNAME=DD3275D,CODE=EBCDIC

A  CONFIG MODEL=2,FEAT=PFK,OPTIONS=NOCOPY
B  CONFIG MODEL=2,OPTIONS=NPGDEL
   LINE  ADDR=0CE,VERIFY=YES,CONFIG=B
   IDLIST (E54C5A,A),E54C5B,(E54C5C,B),(E54C5D,A)

TERMINAL

LINE  ADDR=0CF,VERIFY=NO
IDLIST (E54C5B,A),E54C5C,(E54C5D,B)

TERMINAL

```

VTAM Macro Sets

The following sets of VTAM macro statements describe the VTAM communication devices. A graphic representation of each of these is presented, along with the macro statements required to define it.

COMM Macro Statement

The COMM macro statement can only appear once in an IMS system definition. When VTAM is included in a system, the COMM macro statement is required.

Macro Statement Set Prepared

```

COMM  RECANV=(4,500),APPLID=IMSD,
      OPTIONS=(TIMESTAMP,MFSTEST,FMTMAST,PAGING)

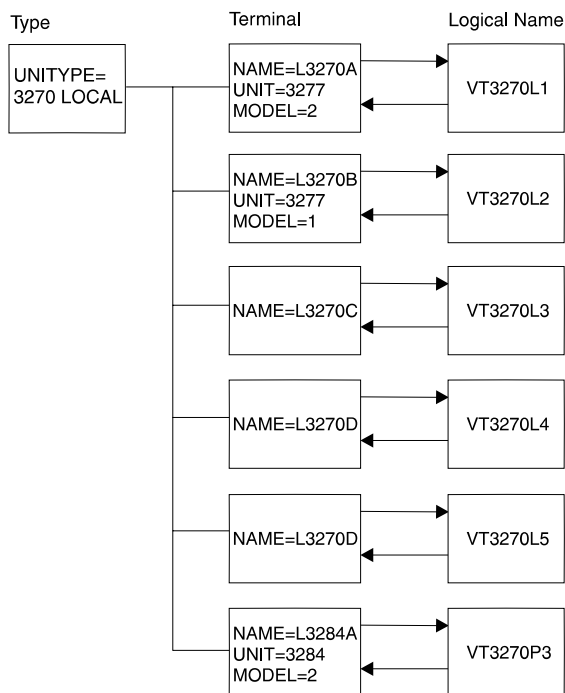
```

Column

72

X

Local 3270 VTAM Terminals

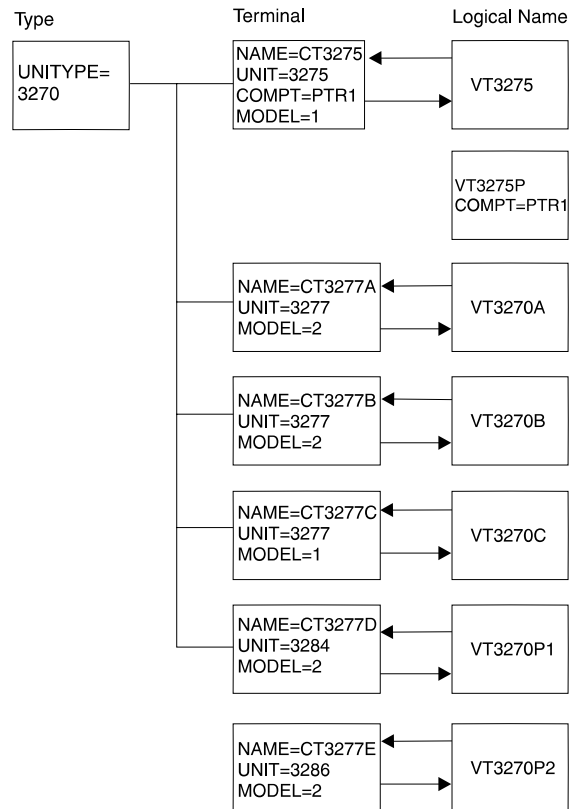


Macro Statement Set Prepared

```

TYPE UNITYPE=(3270,LOCAL),MODEL=2
TERMINAL NAME=L3270A
NAME (VT3270L1,MASTER)
TERMINAL NAME=L3270B,MODEL=1,OPTIONS=FORCRESP
NAME VT3270L2
TERMINAL NAME=L3270C,TYPE=3270-A2,SIZE=(24,80)
NAME VT3270L3
TERMINAL NAME=L3270D,TYPE=3270-A3,SIZE=(32,80)
NAME VT3270L4
TERMINAL NAME=L3270E,TYPE=3270-A4,SIZE=(43,80)
NAME VT3270L5
TERMINAL NAME=L3284A,UNIT=3284,PTRSIZE=132
NAME (VT3270P3,SECONDARY)
    
```

Remote 3270 VTAM Terminals



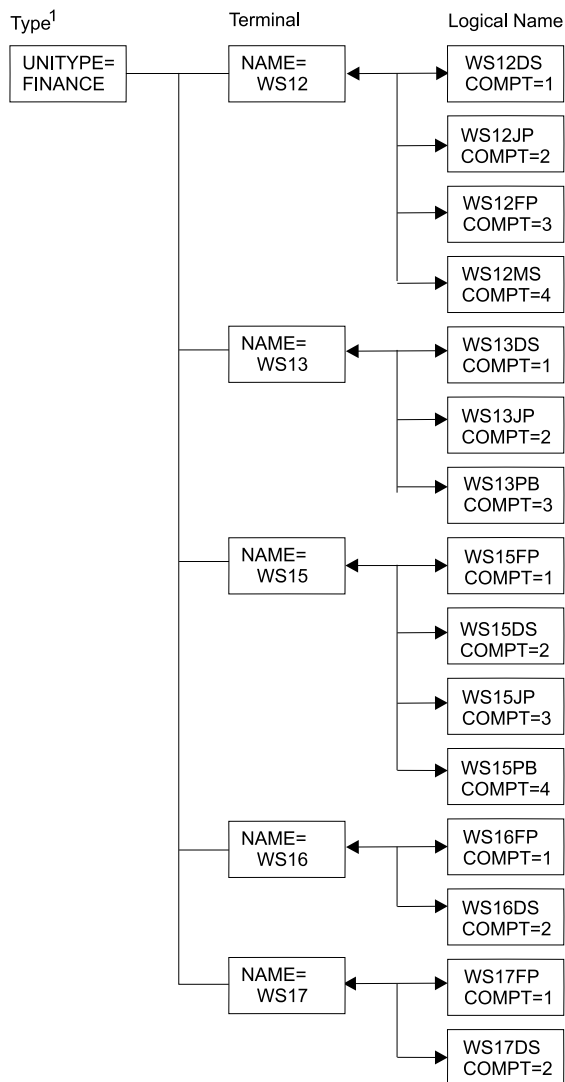
DC Macro Statements

Macro Statement Set Prepared

```

TYPE UNITYPE=3270,MODEL=2,PTRSIZE=132,OPTIONS=COPY
TERMINAL NAME=CT3275,UNIT=3275,COMPT=PTR1,MODEL=1
NAME VT3275
NAME VT3275P,COMPT=PTR1
TERMINAL NAME=CT3277A
NAME VT3270A
TERMINAL NAME=CT3277B
NAME VT3270B
TERMINAL NAME=CT3277C,MODEL=1
NAME VT3270C
TERMINAL NAME=CT3277D,UNIT=3284
NAME VT3270P1
SPACE
TERMINAL NAME=CT3277E,UNIT=3286
NAME VT3270P2
    
```

Finance Communication System



. ¹ Can also specify UNITYPE=3601.

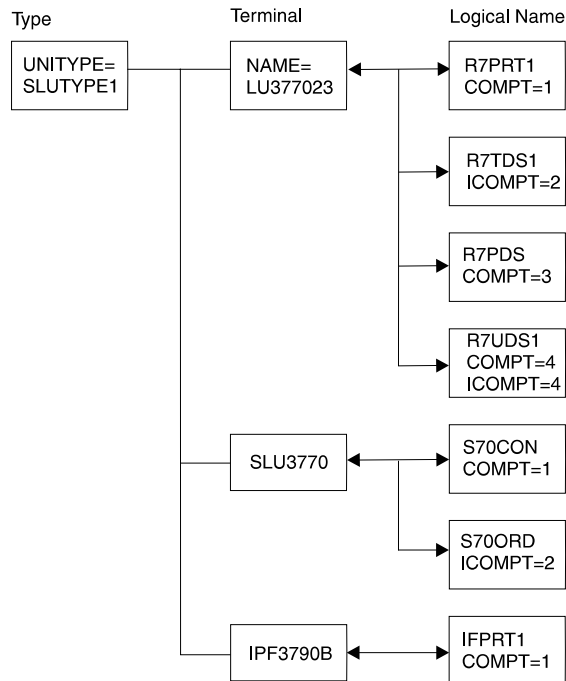
Macro Statement Set Prepared

Column
72

```

TYPE UNITYPE=FINANCE,OUTBUF=256
TERMINAL NAME=WS12,OPTIONS=MFS
      COMPT=(36DS3,36JP,36FP,36MS),FEAT=(DUAL,132)
NAME  WS12DS,COMPT=1
NAME  WS12JP,COMPT=2
NAME  WS12FP,COMPT=3
NAME  WS12MS,COMPT=4
TERMINAL NAME=WS13,OPTIONS=(MFS,NOPNDST,OPTACK),
      COMPT=(36DS4,36JP,36PB)
NAME  WS13DS,COMPT=1
NAME  WS13JP,COMPT=2
NAME  WS13PB,COMPT=3
TERMINAL NAME=WS15,COMPT=(36FP,36DS,36JP,36PB),      X
      OPTIONS=MFS
NAME  WS15FP,COMPT=1
NAME  WS15DS,COMPT=2
NAME  WS15JP,COMPT=3
NAME  WS15PB,COMPT=4
TERMINAL NAME=WS16,COMPT=(36FP,36DS4),
      OPTIONS=(FORCRESP,MFS,OPTACK,BID,OPNDST)
NAME  WS16FP,COMPT=1
NAME  WS16DS,COMPT=2
TERMINAL NAME=WS17,COMPT=(36FP,36DS),
      OPTIONS=(ACK,NOBID,TRANRESP,NOPNDST)
NAME  WS17FP,COMPT=1
NAME  WS17DS,COMPT=2
    
```

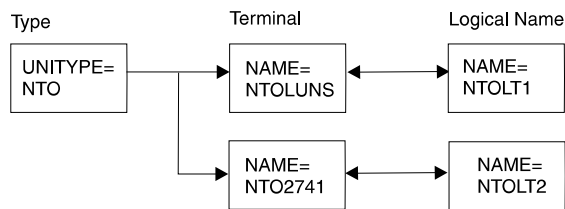
Secondary Logical Unit Type 1 VTAM Terminals



DC Macro Statements

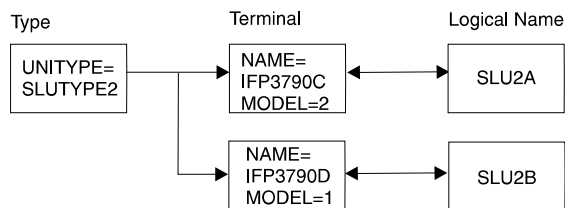
Macro Statement Set Prepared	Column 72
TYPE UNITYPE=SLUTYPE1	
TERMINAL NAME=LU377023,	X
COMPT1=(PRINTER1,BASIC-SCS1),	X
COMPT2=(TRANSDS1,BASIC-SCS2),	X
COMPT3=(PRINTDS1,BASIC-SCS1),	X
COMPT4=(USERDS1,BASIC)	
NAME R7PRT1,COMPT=1	
NAME R7TDS1,ICOMPT=2	
NAME R7PDS1,COMPT=3	
NAME R7UDS1,COMPT=4,ICOMPT=4	
TERMINAL NAME=SLU3770,COMPT1=(CONSOLE,MFS-SCS1),	X
COMPT2=(READER1,MFS-SCS2,3)	
NAME S70CON,COMPT=1	
NAME S70RDR,ICOMPT=2	
TERMINAL NAME=IFP3790B,	X
COMPT1=(PRINTER1,BASIC-SCS1),	X
OPTIONS=(OPNDST,NBSELM)	
NAME IFPRT1,COMPT=1	

NTO Devices



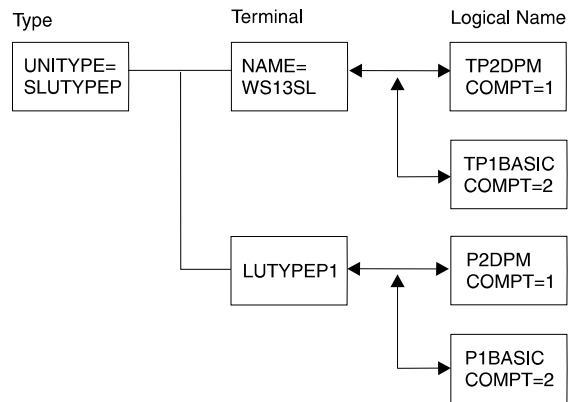
Macro Statement Set Prepared	Column 72
TYPE UNITYPE=NTO,EDIT=(EDIT1,EDIT2)	
TERMINAL NAME=NTOLUNS,EDIT=(YES,YES),	X
OUTBUF=300,SEGSIZE=300,PU=LUNS,OPTIONS=OPNDST	
NAME NTOLT1	
TERMINAL NAME=NTO2741,PU=2741,	X
MSGDEL=NONIOPCB,OPTIONS=(NORESP,MFS)	
NAME NTOLT2	

Secondary Logical Unit Type 2 VTAM Terminals



Macro Statement Set Prepared
TYPE UNITYPE=SLUTYPE2,MODETBL=AAA
TERMINAL NAME=IFP3790C,MODEL=2,MODETBL=BBB
NAME SLU2A
TERMINAL NAME=IFP3790D,MODEL=1,OUTBUF=500
NAME SLU2B

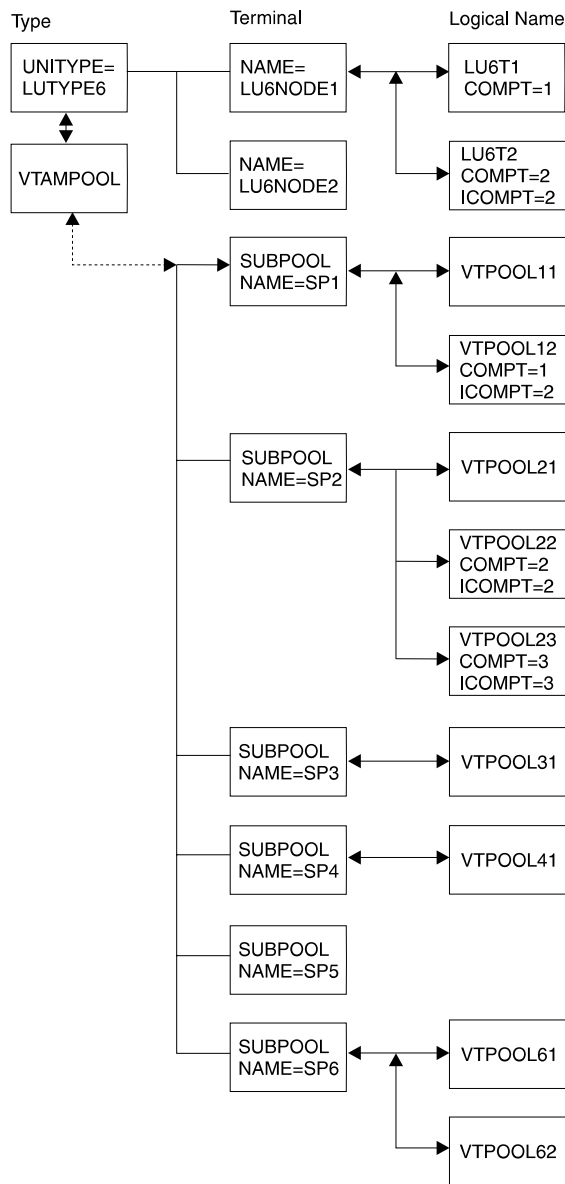
Secondary Logical Unit Type P VTAM Terminals



Macro Statement Set Prepared	Column
	72
TYPE UNITYPE=SLUTYPEP,OUTBUF=256, OPTIONS=(OPTACK,NOBID)	X
TERMINAL NAME=WS13SL, COMPT1=(PROGRAM2,DPM-A1,4), COMPT2=(PROGRAM1,BASIC)	X X
NAME TP2DPM,COMPT=1	
NAME TP1BASIC,COMPT=2	
TERMINAL NAME=LUTYPEP1, COMPT1=(PROGRAM2,DPM-A1,IGNORE), COMPT2=(PROGRAM1,DPM-A3,2), COMPT3=PROGRAM1	X X X
NAME P2DPM,COMPT=1	
NAME P1BASIC,COMPT=2	

DC Macro Statements

Logical Unit Type 6 VTAM Terminals



Macro Statement Set Prepared	Column
	72
TYPE UNITYPE=LUTYPE6	
*LU 6 NODE DEFINITION WITH FIXED LTERM ALLOCATION	
TERMINAL NAME=LU6NODE1,OPTIONS=NORESP,	X
COMPT1=(MULT2,DPM-B1,4),	X
COMPT2=(SINGLE2,VLVB)	
NAME LU6T1,COMPT=1	
NAME LU6T2,COMPT=2,ICOMPT=2	
*LU 6 NODE DEFINITION -- PARALLEL SESSIONS WITH DYNAMIC LTERM ALLOCATION	
TERMINAL NAME=LU6NODE2,OPTIONS=TRANRESP,	X
COMPT1=(SINGLE1,DPM-B1,IGNORE),	X
COMPT2=(SINGLE2,DPM-B3,2),	X
COMPT3=MULT1,	X
SESSION=3	
VTAMPOOL	
SUBPOOL NAME=SP1	
NAME VTPPOOL11	
NAME VTPPOOL12,COMPT=1,ICOMPT=2	
SUBPOOL NAME=SP2	
NAME VTPPOOL21	
NAME VTPPOOL22,COMPT=2,ICOMPT=2	
NAME VTPPOOL23,COMPT=3,ICOMPT=3	
SUBPOOL NAME=SP3	
NAME VTPPOOL31	
SUBPOOL NAME=SP4	
NAME VTPPOOL41	
SUBPOOL NAME=SP5	
SUBPOOL NAME=SP6	
NAME VTPPOOL61	
NAME VTPPOOL62	

System Configuration Macro Statement

In this example, the IMSGEN macro statement must be last in the IMS system definition input stream. The IMSGEN macro statement for that hypothetical system is shown here:.

Multiple Systems Coupling

Characteristics to Define	Label	Macro Statements Prepared	Col. 72
IMS System Configuration		IMSCTRL MSVID=1, IMSID=SYSA IMSCTF	
Databases and Application Macros		DATABASE	
(Application programs that execute in other systems but process transactions entered by System A terminals are defined as remote.)		APPLCTN TRANSACT APPLCTN PSB=A TRANSACT CODE=A APPLCTN PSB=B, SYSID=(2,1) TRANSACT CODE=B APPLCTN PSB=C, SYSID=(3,1) TRANSACT CODE=C	
Data Communication		APPLCTN TRANSACT	
(These logical terminals are defined in Systems B and C as remote logical terminals.)		COMM LINEGRP LINE TERMINAL	
		LINEGRP LINE TERMINAL	
		LINEGRP LINE TERMINAL NAME IMSATRM1 TERMINAL NAME IMSATRM2 TERMINAL NAME IMSATRM3	
Multiple Systems Coupling	LINK1	MSPLINK DDNAME=PLAB1, ADDR=201, TYPE=CTC, BUFSIZE= 1000	X X
	LINK2	MSPLINK TYPE=MTM, BUFSIZE=1000 MSLINK PARTNER=AB, MSPLINK=LINK1	X X
(Terminals in Systems B and C that can enter transaction code A are defined as remote terminals.)	MSC12	MSNAME SYSID=(2,1) NAME IMSBTRM1 NAME IMSBTRM2 NAME IMSBTRM3	
	MSC13	MSNAME SYSID=(3,1) NAME IMSCTRM1 NAME IMSCTRM2 NAME IMSCTRM3 IMSGEN	
		END	

Figure 13. System A IMS System Definition

Multiple Systems Coupling

Characteristics to Define	Label	Macro Statements Prepared	Col. 72
IMS System Configuration		IMSCTRL MSVID=2,IMSID=SYSB IMSCTF	
Databases and Applications (Application programs that execute in other systems but process transactions entered by System B terminals are defined as remote.)		DATABASE APPLCTN TRANSACT APPLCTN PSB=B TRANSACT CODE=B APPLCTN PSB=A,SYSID=(1,2) TRANSACT CODE=A APPLCTN PSB=C,SYSID=(3,2) TRANSACT CODE=C APPLCTN TRANSACT	
Data Communication (These logical terminals are defined in Systems A and C as remote logical terminals.)		COMM LINEGRP LINE TERMINAL LINEGRP LINE TERMINAL LINE LINEGRP LINE TERMINAL NAME IMSBTRM1 TERMINAL NAME IMSBTRM2 TERMINAL NAME IMSBTRM3 TERMINAL NAME IMSVBT1 TERMINAL NAME IMSVBT2 TERMINAL NAME IMSVBT3 TERMINAL NAME IMSVBT4	

Figure 14. System B IMS System Definition (Part 1 of 2)

Multiple Systems Coupling

Multiple Systems Coupling	LINK1	MSPLINK DDNAME=PLBA1, ADDR=302,TYPE=CTC, BUFSIZE=1000	X X
	LINK2	MSPLINK TYPE=MTM, BUFSIZE=1000	X
	LINK3	MSPLINK DDNAME=PLBC1, ADDR=0BA,TYPE=BSC, BUFSIZE=360, CONTROL=YES	X X X
	LINK4	MSPLINK TYPE=VTAM, NAME=NEWYORK, SESSION=2, BUFSIZE=2048	X X X
		MSLINK PARTNER=AB, MSPLINK=LINK1	X
	MSC21	MSNAME SYSID=(1,2) NAME IMSATRM1 NAME IMSATRM2 NAME IMSATRM3	
		MSLINK PARTNER=BC, MSPLINK=LINK3	X
	MSC23	MSNAME SYSID=(3,2) NAME IMSCTRM1 NAME IMSCTRM2 NAME IMSCTRM3	
		MSLINK PARTNER=CB, MSPLINK=LINK4	X
	MSC24	MSNAME SYSID=(4,6) NAME IMSVCT1 NAME IMSVCT2	
		MSLINK PARTNER=CC, MSPLINK=LINK4, OPTIONS=FORCESESS	X X
	MSC25	MSNAME SYSID=(5,7) NAME IMSVCT3 NAME IMSVCT4	
		IMSGEN	
		END	

Figure 14. System B IMS System Definition (Part 2 of 2)

Multiple Systems Coupling

Characteristics to Define	Label	Macro Statements Prepared	Col. 72
IMS System Configuration		IMSCTRL MSVID=3,IMSID=SYSC IMSCTF	
Databases and Applications		DATABASE	
(Application programs that execute in other systems but process transactions entered by System C terminals are defined as remote.)		APPLCTN TRANSACT APPLCTN PSB=C TRANSACT CODE=C APPLCTN PSB=A,SYSID=(1,3) TRANSACT CODE=A APPLCTN PSB=B,SYSID=(2,3) TRANSACT CODE=B	
Data Communication		APPLCTN TRANSACT	
(These logical terminals are defined in Systems A and B as remote logical terminals.)		COMM LINEGRP LINE TERMINAL LINEGRP LINE TERMINAL LINE LINEGRP LINE TERMINAL NAME IMSCTRM1 TERMINAL NAME IMSCTRM2 TERMINAL NAME IMSCTRM3 TERMINAL NAME IMSVCT1 TERMINAL NAME IMSVCT2 TERMINAL NAME IMSVCT3 TERMINAL NAME IMSVCT4	

Figure 15. System C IMS System Definition (Part 1 of 2)

Data Sharing Sys Configuration

```

Multiple
Systems
Coupling
LINK1 MSPLINK DDNAME=PLCB1,      X
      ADDR=0BB,TYPE=BSC,        X
      BUFSIZE=360,CONTROL=NO
LINK4 MSPLINK TYPE=VTAM,        X
      NAME=HARTFORD,            X
      SESSION=2,                X
      BUFSIZE=2048

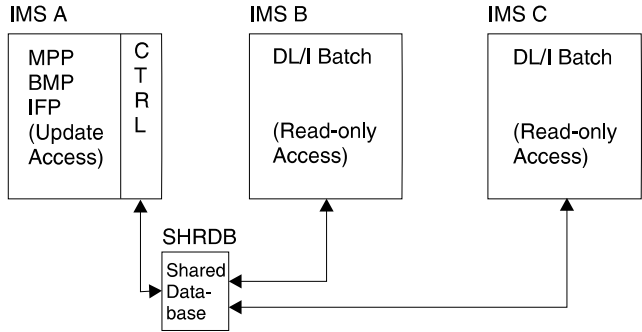
      MSLINK PARTNER=BC,        X
      MSLINK=LINK1
MSC31 MSNAME SYSID=(1,3)
      NAME IMSATRM1
      NAME IMSATRM2
      NAME IMSATRM3
MSC32 MSNAME SYSID=(2,3)
      NAME IMSBTRM1
      NAME IMSBTRM2
      NAME IMSBTRM3
      MSLINK PARTNER=CB,        X
      MSLINK=LINK4
MSC34 MSNAME SYSID=(6,4)
      NAME IMSVBT1
      NAME IMSVBT2
      MSLINK PARTNER=CC,        X
      MSLINK=LINK4,            X
      OPTIONS=FORCESESS
MSC35 MSNAME SYSID=(7,5)
      NAME IMSVBT3
      NAME IMSVBT4
      MSGEN
      END
  
```

Figure 15. System C IMS System Definition (Part 2 of 2)

Data-Sharing System Configuration

The following examples show four possible data-sharing system configurations. The IMS system definition macros that must be defined to support these configurations and the associated PSB generation statements are shown for each configuration. The illustrations below assume that the database shown is registered with DBRC.

Example 1. Data Sharing at the Database-Level with Update Access



System A

IMS system definition statements include:
 IMSCTRL IMSID=IMSA,DBRC=YES

Data Sharing Sys Configuration

```
DATABASE DBD=SHRDB,ACCESS=UP
```

PSB generation statements include:

```
PCB TYPE=DB,DBDNAME=SHRDB,PROCOPT=A
```

System B

IMS system definition statements include:

```
IMSCTRL IMSID=IMSB,DBRC=YES,SYSTEM=(VS2,BATCH)
```

PSB generation statements include:

```
PCB TYPE=DB,DBDNAME=SHRDB,PROCOPT=GO
```

System C

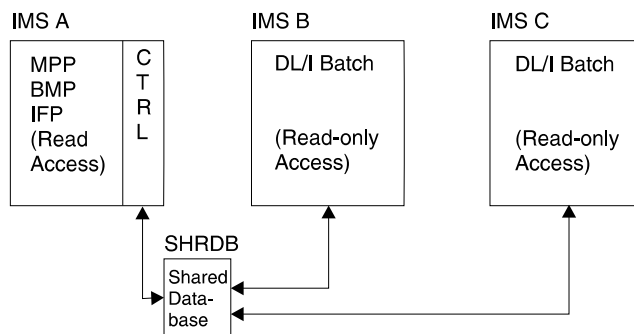
IMS system definition statements include:

```
IMSCTRL IMSID=IMSC,DBRC=YES,SYSTEM=(VS2,BATCH)
```

PSB generation statements include:

```
PCB TYPE=DB,DBDNAME=SHRDB,PROCOPT=GO
```

Example 2. Data Sharing at the Database Level with Read Access



System A

IMS system definition statements include:

```
IMSCTRL IMSID=IMSA,DBRC=YES
```

```
DATABASE DBD=SHRDB,ACCESS=RD
```

PSB generation statements include:

```
PCB TYPE=DB,DBDNAME=SHRDB,PROCOPT=G
```

System B

IMS system definition statements include:

```
IMSCTRL IMSID=IMSB,DBRC=YES,SYSTEM=(VS2,BATCH)
```

PSB generation statements include:

```
PCB TYPE=DB,DBDNAME=SHRDB,PROCOPT=GO
```

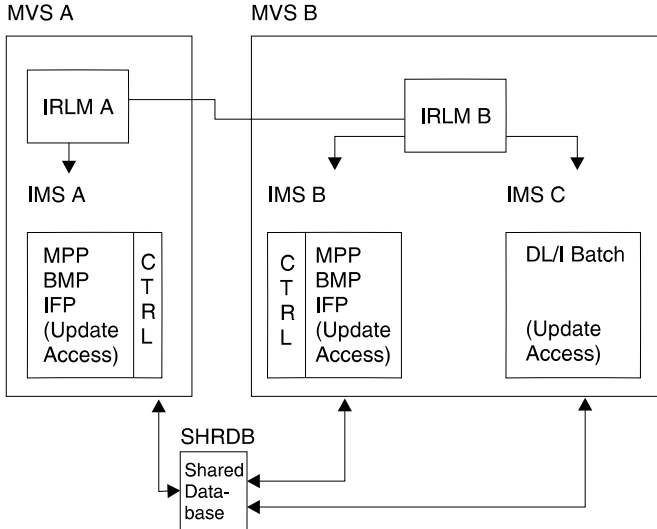
System C

IMS system definition statements include:

```
IMSCTRL IMSID=IMSC,DBRC=YES,SYSTEM=(VS2,BATCH)
```

PSB generation statements include:
 PCB TYPE=DB,DBDNAME=SHRDB,PROCOPT=G

Example 3. Intra-CPC Block-Level Data Sharing



System A

IMS system definition statements include:
 IMSCTRL IMSID=IMSA,DBRC=YES,IRLMNM=RLMA
 DATABASE DBD=SHRDB,ACCESS=RD

PSB generation statements include:
 • PCB TYPE=DB,DBDNAME=SHRDB,PROCOPT=A

System B

IMS system definition statements include:
 IMSCTRL IMSID=IMSB,DBRC=YES,IRLMNM=RLMA, X
 SYSTEM=(VS2,BATCH)

PSB generation statements include:
 PCB TYPE=DB,DBDNAME=SHRDB,PROCOPT=GO

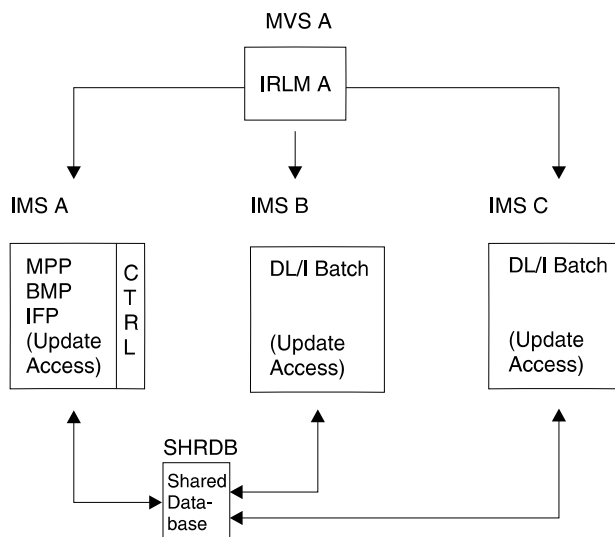
System C

IMS system definition statements include:
 IMSCTRL IMSID=IMSC,DBRC=YES,SYSTEM=(VS2,BATCH)

PSB generation statements include:
 IMSCTRL IMSID=IMSC,DBRC=YES,IRLMNM=RLMA, X
 SYSTEM=(VS2,BATCH)
 PCB TYPE=DB,DBDNAME=SHRDB,PROCOPT=A

Data Sharing Sys Configuration

Example 4. Inter-CPC Block-Level Data Sharing



System A

IMS system definition statements include:

```

IMSCTRL  IMSID=IMSA,DBRC=YES,IRLMNM=RLMA
DATABASE DBD=SHRDB,ACCESS=UP
  
```

PSB generation statements include:

- PCB TYPE=DB,DBDNAME=SHRDB,PROCOPT=A

System B

IMS system definition statements include:

```

IMSCTRL  IMSID=IMSB,DBRC=YES,IRLMNM=RLMB,      X
DATABASE DBD=SHRDB,ACCESS=UP
  
```

PSB generation statements include:

```

PCB      TYPE=DB,DBDNAME=SHRDB,PROCOPT=A
  
```

System C

IMS system definition statements include:

```

IMSCTRL  IMSID=IMSC,DBRC=YES,IRLMNM=RLMB,      X
SYSTEM=(VS2,BATCH)
  
```

PSB generation statements include:

```

PCB      TYPE=DB,DBDNAME=SHRDB,PROCOPT=A
PCB      TYPE=DB,DBDNAME=SHRDB,PROCOPT=A
  
```

IMS DBCTL Environment

The following shows sample stage 1 input specifications for the DBCTL environment:

DBCTL Environment

```

Column
72
IMSCTRL SYSTEM=(VS/2,(ON-LINE,DBCTL),4.2), X
MAXREGN=(20,52K,A,A), X
MCS=(2,7),DESC=7,MAXCLAS=1,IMSID=IMSA, X
CMDCHAR=4
*
IMSCTF SVCNO=(,203,202), X
LOG=(DUAL,MONITOR), X
RDS=(3380,4096), X
CPLOG=1000
*
* DEFINE SYSTEM BUFFERS
*
* BUFPOOLS PSBW=60000,DMB=10000, X
SASPSB=(20000,80000)
*
* DEFINE FAST PATH OPTIONS
* FPCTRL OTHREAD=10,BFALLOC=(5,20,1024)
*
* DEFINE SECURITY OPTIONS
* SECURITY TYPE=AGNEXIT
*
* DEFINE DL/I DATABASES
*
* DATABASE RESIDENT,DBD=DI21PART
*
* DEFINE SAMPLE APPLICATIONS
*
* APPLCTN PSB=DFHSAM04,SCHDTYP=PARALLEL
* APPLCTN PSB=DFHSAM05,SCHDTYP=PARALLEL
*
* <!-- *****KVR0016***** -->
IMSGEN ASM=(HSAM,SYSLIN), X
ASMPRT=ON, X
LKPRT=(XREF,LIST), X
LKSIZE=(880K,64K), X
LKRGN=4096K, X
SUFFIX=1, X
SURVEY=NO, X
SYMSMG=TIMESTAMP, X
MACLIB=UTILITY, X
OBJDSET=IMS.OBJDSET, X
USERLIB=IMS.LOADLIB, X
PROCLIB=(YES,), X
NODE=(IMS,IMS,IMS), X
JCL=(GENJOB, X
(1), X
PGMERID, X
A, X
(TIME=5,CLASS=K,NOTIFY=PGMERID)), X
SCL=(99)
END

```

DBCTL Environment

Appendix B. MFS System Definition and Programming Considerations

For a complete description of IMS system definition macros, see “Chapter 3. Macros” on page 53.

Related Reading: For additional information on installation requirements, see *IMS/ESA Installation Volume 1: Installation and Verification*.

System Definition Macro Instructions

The system definition macro instructions that include specifications for MFS are summarized below.

BUFPOOLS Macro

BUFPOOLS is used to define

- The size of the MFS buffer pool
- The number of fetch request elements to be used for loading MFS control blocks into the MFS buffer pool (with the FORMAT=, COMM=, and FRE= keywords).

Related Reading: For formulas used in calculating the buffer pool sizes, see *IMS/ESA Administration Guide: System*. Default values are calculated and explained in “Chapter 3. Macros” on page 53. The pool size values set at system definition are used unless other values are specified at initialization of the online system control region.

The following I/O pool requirements for MFS are provided by the MFS language utility using message DFS1060I:

- Largest line buffer size for 3270 displays
- Largest work buffer size for an input message
- Largest work buffer size for an output message

For efficient operation in the virtual control region, the areas acquired for buffer pools during initialization can be fixed in address space during initialization.

MSGQUEUE Macro

MSGQUEUE is used to specify the maximum size output segment that will be properly processed by MFS.

IMSGEN Macro

IMSGEN is used to specify MFSTEST parameters if the COMM macro is not used. The SUFFIX= operand specifies the suffix of the MFS device characteristics table that is generated when TYPE or TERMINAL macros define device symbolic names (that is, TYPE=3270-An) during system definition.

COMM Macro

COMM is used to specify the following facilities of MFS:

- Exits for user-written field and segment editing routines to be included in the generated system
- IMS-provided formatting for MFS on the 3270 or SLU 2 master terminal

MFS System Definition

The IMS-provided formatting for the master terminal is provided if `OPTIONS=(...,FMTMAST,...)` is specified and the 3270 or SLU 2 display used as the master terminal is a 24x80 screen defined as `TYPE=(3270,2)` or `3270-An` with `SIZE=(24,80)`.

The name used for the `EDITNAME` parameter cannot be the same as any MID name.

TYPE, TERMINAL, and CONFIG Macros

`TYPE`, `TERMINAL`, and `CONFIG` are used to describe terminal features.

The terminals that can be defined to operate with MFS by their number or by secondary logical unit type are described in "Chapter 1. Designing the IMS System" on page 1.

TYPE Macro

The `TYPE`, `SIZE`, `FEAT`, and `OPTIONS` keywords can be used in the `TYPE` and `TERMINAL` macro. The `OPTIONS=` operand of the `TYPE` macro is used to request MFS support for a set of terminal devices described in the subsequent `TERMINAL` macros. If 3270 or SLU 2 terminals are defined, MFS support is automatic. The `TYPE`, `SIZE`, and `FEAT` keywords define the 3270 and SLU 2 devices using the device symbolic name and generate the MFS device characteristics table containing the device type symbolic name, associated screen size, and physical terminal features. `TYPE` keywords provide definition for subsequent `TERMINAL` macros.

For NTO, specify `UNITYTYPE=NTO` on the `TYPE` macro.

TERMINAL Macro

The `COMPTn=` operand can be used to request MFS support for a specific component of a terminal defined as a secondary logical unit.

For NTO, specify `PU=xx` where `xx` is `LUNS`, `2741`, `2740-1`, or `TTY`.

The `COMPTn=` operand of the `TERMINAL` macro can request MFS support for the secondary logical units as follows:

For `TYPE UNITYTYPE=SLUTYPE1`: `COMPTn=(...,MFS-SCS1,...)` must be specified for the MFS SCS1 formatting option for the console, printer, or print data set component.

`COMPTn=(...,MFS-SCS2,...)` must be specified for the MFS SCS2 formatting option for the reader, punch, or transmit data set component.

For `TYPE UNITYTYPE=SLUTYPEP`: `COMPTn=(...,DPM-An,...)` must be specified for the MFS DPM formatting option for the remote controller component.

`COMPTn=(...,MFS-SCS1,...)` must be specified for the MFS SCS1 formatting option for the remote controller component.

For `TYPE UNITYTYPE=LUTYPE6`: `COMPTn=(...,DPM-Bn,...)` must be specified for MFS ISC formatting.

Related Reading: See *IMS/ESA Customization Guide* for MFS definition examples for ISC.

MFS System Definition

The following is a sample definition that would allow DPM-An to be used with a remote program. The definition is for a secondary logical unit with two components.

```
TYPE          UNITYPE=SLUTYPEP

TERMINAL      NAME=LUX,OPTIONS=(FORCRESP,ACK,NOBID),
              OUTBUF=256,COMPT1=(PROGRAM2,DPM-A3,7),
              COMPT2=(PROGRAM1,DPM-A4,1)

NAME          LTERMQ,COMPT=1

NAME          LTERMY,COMPT=2
```

PROGRAM1 is printed output operation; PROGRAM2 is paged output operation.

The OUTBUF= specification would allow RCDCTL to be up to 256 bytes in all format definitions for this secondary logical unit.

The following is a sample definition that would allow DPM-An or SCS1 to be used with a remote program. For this secondary logical unit, the DPM-An definition is for the first two components, and the SCS1 definition is for the third component.

```
TYPE          UNITYPE=SLUTYPEP

TERMINAL      NAME=LUX,OPTIONS=(FORCRESP,ACK,NOBID),
              OUTBUF=256,COMPT1=(PROGRAM2,DPM-A3,7),
              COMPT2=(PROGRAM1,DPM-A4,1),
              COMPT3=(PROGRAM1,MFS-SCS1,IGNORE)

NAME          LTERMQ,COMPT=1

NAME          LTERMY,COMPT=2

NAME          LTERMS,COMPT=3
```

PROGRAM1 is an unprotected operation at the end of a message for paged output or a printer (SCS1) operation; PROGRAM2 is a protected operation at the end of a message for paged output.

The OUTBUF= specification would allow RCDCTL to be up to 256 bytes in all DPM format definitions for this secondary logical unit.

If OUTBUF is not large enough for an entire data page, then more than one VTAM SEND is required to send the page.

For TYPE UNITYPE=NTO: OPTIONS=(...,MFS,...) must be specified on the TERMINAL macro for the MFS SCS1 formatting option to be used by the NTO device.

The following is a sample definition that would allow MFS to be used by an NTO device. MFS uses the SCS1 formatting option and treats the NTO device as an SLU 1 console. In this example, the NTO device is an IBM 3101 (LUNS).

```
TYPE          UNITYPE=NTO
TERMINAL      NAME=LU3101,PU=LUNS,
              OPTIONS=(TRANRESP,MFS,OPNDST),SEGSIZE=300
NAME          A3101
```

The TYPE, SIZE, FEAT, and OPTIONS keywords described in the TYPE macro above can be used in the TERMINAL macro.

MFS System Definition

The `TERMINAL` or `TYPE` macro indicates to MFS whether or not operator input that invokes processing for a new transaction should dequeue the output message for the current transaction. The parameters that apply are `OPTIONS=PAGDEL` or `NPAGDEL` or `PAGDEL=YES` or `NO`. Refer to the section, "TERMINAL Macro" on page 140 for a discussion of the effects of these specifications.

CONFIG Macro

The `CONFIG` macro describes the configuration for a switched 3275 terminal.

System Programming Considerations

The major system programming considerations in implementing MFS are summarized below.

INDEX Directory

The `INDEX` function of the MFS service utility creates an index directory to specified control blocks in the staging library, `IMS.FORMAT`. The IMS initialization process uses the index directory to construct the MFS dynamic directory in a private storage area. This space is separate from the MFS buffer pool.

If a requested control block is not found in the MFS buffer pool, MFS looks for an entry in the MFS dynamic directory. If found, the entry gives the relative track address for the indexed control block. This allows MFS to issue a direct read without going through the active format library's PDS directory. If the entry is not found, MFS reads the PDS directory, then reads the requested control block. MFS also creates an entry in the MFS dynamic directory for the control block.

The number of entries in the MFS dynamic directory continually increases until the MTO uses the command `/CHANGE DIRECTORY MFS`. This purges all the entries that were added to the MFS dynamic directory after online initialization.

Error conditions include:

- IMS abending during initialization if storage is unavailable for the MFS dynamic directory
- The `/MODIFY COMMIT` command failing during online change if storage is unavailable for the MFS dynamic directory or the PDS directory index
- No entries being added to the MFS dynamic directory during online operation if storage is unavailable for the MFS dynamic directory. To add entries to the MFS dynamic directory again, issue a `/CHANGE DIRECTORY MFS` command, or restart IMS.

FORMAT, FORMATA, FORMATB, and TFORMAT Library Directories

Each PDS directory entry in the format libraries contains 14 bytes if created by a release of IMS prior to Release 1.6, or 24 bytes if created by IMS Release 1.6 or later. The 14-byte entry contains 12 bytes for the block name and TTR, and 2 bytes for the block size. The 24-byte entry adds 10 bytes for a data and time stamp and expansion of the block size to 4 bytes. IMS is able to read a directory containing a mixture of 14-byte and 24-byte entries.

Use of Concatenated Format Libraries

You can define several format libraries on different physical or logical devices, thus permitting faster (concurrent) input/output processing. IMS limits the number of data sets that can be concatenated for MFS format libraries, `IMS.FORMATA` and `IMS.FORMATB` to 16. All of the concatenations must have like attributes.

System Programming Considerations for MFS

Phase 2 of the MFS language utility implicitly deletes an old control block from a format library when a new control block with the same name is defined for that library. However, if the new control block is defined for a different library than the one the old copy is in, there will be no implicit deletion. Thus, old and new copies for control blocks can exist simultaneously in different format libraries.

One way to prevent this duplication is to explicitly delete duplicate control blocks, using the MFS service utility SCRATCH function. Another way is to add a DD statement in the MFSUTIL phase 2 JCL for each format library (up to 15) that is to have old control blocks deleted when new versions of those control blocks are defined for the staging library, IMSVS.FORMAT. Phase 2 will make the deletions, and will update the index directory \$\$IMSDIR (if there is one) in each library to reflect the new contents.

The new DD statements must have labels of the form, FORMATn, where n is a number from 1 to 15, inclusive. Phase 2 will only compress and add control blocks to the format library defined with a DD statement labeled FORMAT. (FORMAT is the original staging library.)

At IMS online initialization, the index directories (\$\$IMSDIR) of each of the format libraries are merged into a single MFS dynamic directory for all the format libraries. If a requested control block is not listed in the MFS dynamic directory, MFS will read one PDS directory block from each concatenated library, in sequence, until it finds the control block. This can mean many additional reads if the control block resides on one of the last concatenated libraries.

Use of the MFS Buffer Pool

MFS control blocks are loaded into the MFS buffer pool by one of two MFS pool manager routines: an immediate fetch routine or a prefetch routine. The immediate fetch routine loads a control block that is needed immediately and not already in storage. The prefetch routine is used only if the prefetch option is specified during initialization of the online IMS control region. It is invoked when IMS anticipates the need for a control block; if the block is already in storage, use of prefetch lowers the likelihood that the block will be forced out to make room for another block. The use of prefetch requests can improve response time.

When a requested control block is not already in the pool, and its name resides in the resident directory, \$\$IMSDIR, a single read operation loads the control block into storage. If the name is not in the resident directory, a read of one data set directory block followed by a read of the control block loads the control block into storage. Multiple reads of the data set occur when the size of the control block is greater than either the track size or the MVS-specified BLKSIZE.

Fetch Request Elements

Fetch request elements (FREs) are used to represent control blocks that are either already in storage or have been requested for loading into storage. The number of FREs to be defined depends on the number of control blocks that can reside in the pool (a function of pool size) and the number of concurrent requests to load control blocks. FREs defined during IMS system definition or during initialization of the online IMS control region are allocated contiguously in the buffer pool.

Pool Space Management

The main objectives of MFS buffer pool space management are:

- To keep as many control blocks as possible available in the pool
- To minimize pool fragmentation

System Programming Considerations for MFS

The MFS pool manager assigns a priority to each fetch request. FREs represent prefetch requests and immediate fetch requests. When required, additional FREs are assigned from available pool space to represent immediate fetch requests. If an FRE is not available when a fetch request is made, a control block of lesser importance is discarded so that its FRE can be used to represent the new request.

When there is not enough pool space to hold a requested control block, a space-freeing algorithm is used to select less important control blocks (based on their frequency of use) for discarding.

Pool Statistics

Use the `/DISPLAY POOL` command to display the contents of the MFS buffer pool counters. The statistics displayed for directory and fetch input/output operations and pool space use should help in determining the appropriate pool size and number of FREs.

Related Reading: See *IMS/ESA Administration Guide: System* for formulas to calculate the format buffer pool size and the number of FREs.

Bibliography

This bibliography includes all the publications cited in this book, including the publications in the IMS library.

3274 Control Unit Description and Programmer's Guide, GA23-0061

Advanced Communications Function for VTAM (ACF/VTAM) Macro Language Guide, SC38-0256

Advanced Communications Function for VTAM (ACF/VTAM) Macro Language Reference, SC38-0261

CICS Extended Recovery Facility Guide, SC33-0661

CICS Installation Guide, SC33-0663

CICS Operations Guide, SC33-0668

CICS Resource Definition (Macro), SC33-0667

CICS System Definition Guide, SC33-0664

IBM DATABASE 2 Version 2 Administration Guide, SC26-4374

IBM DATABASE 2 Version 2 SQL Reference, SC26-4380

IMS/ESA CICS-IMS Database Control Guide, SC33-1184

MVS/ESA Conversion Notebook for System Product Version 3, Vols. 1 and 2, GC28-1567 and GC28-1568

MVS/ESA Data Administration: Utilities, SC26-4516

MVS/ESA Data Facility Product Version 3: Utilities, GC26-4559

MVS/ESA Interactive Problem Control System (IPCS) Planning and Customization, GC28-1832

MVS/ESA Interactive Problem Control System (IPCS) Users Guide, GC28-1833

MVS/ESA Job Control Language Reference, GC28-1829

MVS/ESA Job Control Language User's Guide, GC28-1830

MVS/ESA Linkage Editor and Loader User's Guide, SC26-4510

MVS/ESA Service Aids, GC28-1844

MVS/ESA System Programming Library: Application Development Guide, GC28-1852

OS PL/I Version 2: Programming Guide, Release 3, SC26-4307

Programming Language Considerations in an Information Management System (IMS) Environment, G320-9538

VS Cobol II Application Programming Guide, SC26-4045

IMS/ESA Version 6 Library

SC26-8725	ADB	Administration Guide: Database Manager
SC26-8730	AS	Administration Guide: System
SC26-8731	ATM	Administration Guide: Transaction Manager
SC26-8727	APDB	Application Programming: Database Manager
SC26-8728	APDG	Application Programming: Design Guide
SC26-8726	APCICS	Application Programming: EXEC DLI Commands for CICS and IMS
SC26-8729	APTM	Application Programming: Transaction Manager
SC26-8732	CG	Customization Guide
SC26-9517	CQS	Common Queue Server Reference
SC26-8733	DBRC	Database Recovery Control Guide and Reference
LY37-3731	DGR	Diagnosis Guide and Reference
LY37-3732	FAST	Failure Analysis Structure Tables (FAST) for Dump Analysis
GC26-8736	IIV	Installation Volume 1: Installation and Verification
GC26-8737	ISDT	Installation Volume 2: System Definition and Tailoring
SC26-8740	MIG	Master Index and Glossary
GC26-8739	MC	Messages and Codes
SC26-8743	OTMA	Open Transaction Manager Access Guide
SC26-8741	OG	Operations Guide
SC26-8742	OR	Operator's Reference
GC26-8744	RPG	Release Planning Guide
SC26-8767	SOP	Sample Operating Procedures
SC26-8769	URDB	Utilities Reference: Database Manager
SC26-8770	URS	Utilities Reference: System
SC26-8771	URTM	Utilities Reference: Transaction Manager

Supplementary Publications

GC26-8738	LPS	Licensed Program Specifications
-----------	-----	---------------------------------

Bibliography

Supplementary Publications

SC26-8766 SOC Summary of Operator
Commmands

Online Softcopy Publications

LK3T-2326 CDROM IMS/ESA Version 6 Softcopy
Library
SK2T-0730 CDROM IBM Online Library: Transaction
Processing and Data
SK2T-0710 CDROM MVS Collection
SK2T-6700 CDROM OS/390 Collection

Index

Numerics

- 2305 device 81, 117
- 2740 terminal
 - fast path terminal 140
 - LINE macro statement, specifying 108
 - LINEGRP macro statement, specifying 112
 - master terminal devices, choosing 22
 - system definition example 427, 436, 440
 - TERMINAL macro statement, specifying 140
- 2741 terminal
 - LINEGRP macro statement, specifying 112
 - system definition example 427, 437, 441
 - TERMINAL macro statement, specifying 140
- 2780 terminal
 - LINEGRP macro statement, specifying 112
 - TERMINAL macro statement, specifying 140
- 2848 control unit 74
- 2972 control unit 74
- 3270 terminal
 - LINE macro statement, specifying 108
 - LINEGRP macro statement, specifying 112
 - master terminal devices, choosing 22
 - TERMINAL macro statement, specifying 140
- 3271 control unit 74
- 3275 terminal
 - LINE macro statement, specifying 108
 - LINEGRP macro statement, specifying 112
- 3340 device 116
- 3350 device 116
- 3375 device 116
- 3380 device 116
- 3390 device 116
- 3600 work station 140

A

- ACBGEN utility, PSB pools and 6, 64
- APPLCTN macro
 - description 56
 - dynamic reassignment 58
 - MSNAME macro, and 60
 - parameters 58
 - syntax diagram 56
 - TRANSACT macro, and 56, 59, 60
- application programs
 - online applications, defining 9
 - program resource requirements, defining 56
- application programs, OS/390
 - using the ODBA interface 411

B

- batch dynamic allocation, disabling 360
- BMP
 - DB2, and 419
 - declaring 12
 - EXEC parameters 375

- buffer pool
 - MFS 465
 - OSAM 378, 379
 - OSAM SB 378
 - sizes 365
 - specifying 378
 - VSAM 378, 379
- BUFPOOLS macro
 - description 61
 - MFS considerations 461
 - parameters 62
 - syntax diagram 62

C

- CBLTDLI procedure 243
- CCTL
 - DBCTL databases, and 395
 - preparing 396
 - starting 396
- checkpoint
 - data sets 284
 - frequency of, setting 3
- CHECKPOINT 4, 127
- CICS
 - accessing IMS databases 415
 - data sharing 415
 - DBCTL, and 417
 - DL/I, and 415, 417
 - intersystem communication, and 416
 - IRLM, and 415
 - PSB coding considerations 416
 - sequential buffering, and 417
 - VTAM, and 415
- COBOL programs, coding 382
- COMM macro
 - description 64
 - IMSGEN macro, and 137
 - MFS, and 461
 - parameters 65
 - SECURITY macro, and 137
 - syntax diagram 64
 - TERMINAL macro, and 72
 - VTAM terminals, and 64, 66
- CONFIG macro
 - description 73
 - IDLIST macro, and 73
 - LINE macro, and 73
 - LINEGRP macro, and 73
 - MFS 464
 - parameters 73
 - syntax diagram 73
- control unit
 - 2848 74
 - 2972 74
 - 3271 74
 - CTLUNIT macro 74

- coupling facility
 - OSAM data caching 362
 - structure names for Sysplex data sharing 360
- CQS
 - address space 336
 - shared queues, placing messages on 273
- CTLUNIT macro
 - description 74
 - parameters 74
 - syntax diagram 74

D

- DASD
 - log tracing 353
 - logging 279, 280, 281, 358
- data communication
 - EXEC parameters 372, 376
 - intelligent stations, defining 17
 - macros 434
 - master terminal, specifying 21
 - switched communication devices, defining 18
 - terminals, defining
 - non-VTAM terminals 14, 15
 - VTAM terminals 14, 19, 20
- data sharing
 - DATABASE macro statement 9
 - IMSCTRL macro statement 9
 - system configuration example 455
- DATABASE macro
 - CICS, and 417
 - description 75
 - online applications, defining 9
 - parameters 75
 - syntax diagram 75
- databases
 - batch, tracking 193, 212
 - CICS, accessing with 415
 - DB2, accessing 419
 - DBRC, registering with 230, 253
 - DL/I, using 195, 216
 - example 427
 - Fast Path 252, 256
 - main storage (MSDBs) 304
 - managing 75
 - online, declaring 9
- DB2
 - BMPs, and 419
 - DB2 tasks 419
 - Fast Path, and 419
 - IMS, and 419
 - IMS batch, and 424
 - IMS.PROCLIB, and 419
 - IMS tasks, and 419
 - language interface module, defining 423
 - message processing programs, and 419
 - modules 425
 - SSM parameter, defining 422
 - subsystem connection, tracing 356
 - subsystem identification parameters 369
 - subsystems, defining 419
- DB2 (*continued*)
 - tables, providing 425
 - trace options, specifying 424
- DBBBATCH procedure 192
- DBC procedure
 - description 195
 - storing 191
- DBCTL
 - CCTL, and 395
 - CICS, and 417
 - environment, example 458
 - macros that apply 30
 - using 30
- DBFMSDBx
 - description 304
 - Main Storage Databases, specifying 304
 - parameters 304
- DBRC
 - DATABASE macro, ACCESS parameter 75
 - procedure, description 200
 - procedure, storing 191
- DC macro
 - examples 434
 - macros 434
- DCC procedure
 - description 201
 - storing 191
- DCCTL
 - macros that apply 30
 - using 30
- DFS62DTx
 - description 363
 - parameters 364
- DFSDCxxx
 - data communications options, specifying 305
 - description 305
 - parameters 305
- DFSDFRnn
 - description 306
 - parameters 307
- DFSFDRxx
 - description 307
 - parameters 307
- DFSFIxnn
 - description 309
 - parameters 309
- DFSHSBxx
 - description 313, 318
 - parameters 313, 318
 - preinitialization routines, specifying dependent region 318
 - XRF options 313
- DFSINTxx
 - description 318
 - parameters 318
- DFSMPLEXx
 - description 318
 - high-use program modules, making resident 318
 - parameters 319
- DFSMPR procedure 211
 - description 211

- DFSMPR procedure 211 *(continued)*
 - IMSMSG job 237
- DFSPBDBC
 - description 320
 - parameters 320
- DFSPBDCC
 - description 320
 - parameters 320
- DFSPBIMS
 - description 321
 - parameters 321
- DFSRESLIB DD statement
 - IMS procedures, specifying in 192
- DFRSRxx
 - description 321
 - parameters 322
- DFSSPMxx
 - description 332
 - examples 336
 - parameters 333
- DFSSQxxx
 - description 336
 - parameters 337
- DFSSTAT report 280
- DFSVFLqq
 - description 338
 - parameters 339
- DFSVSMxx
 - control statements, types of 339
 - description 339
 - Fast Path DEDB buffer pools, defining 340
 - OSAM buffer pools, defining 347
 - OSAM subpools, defining 347
 - sequential buffering, specifying 350
 - serviceability and trace options, defining 350
 - VSAM buffer pools, defining 341
 - VSAM performance options 346
 - VSAM subpools, defining 342
- DISPLAY command 6
- DL/I
 - accounting procedures 7
 - call image trace 357
 - CICS, and 415, 417
 - COBOL interface 243
 - DBC procedure, and 195
 - DLISAS procedure, modifying 6
 - execution parameters, system 368
 - exits, modifying 7
 - interfaces 243
 - invoking 6
 - lock activity, tracing 354
 - PL/I interface 243
 - security considerations 7
 - selecting 5
 - storage considerations 5
 - tuning considerations 7
- DLIBATCH procedure 212
- DLISAS procedure
 - description 214
 - IMS procedure, and 221
 - storing 191

- DRA start-up table
 - using with ODBA 411
- DREF storage, defining 307
- dumping
 - batch system 291
 - dependent region 291
 - installing 287
 - IRLM, and 293
 - MVS options 288
 - online 288
 - options 287
 - system execution parameters 369
- DXRJPROC procedure 216
- dynamic reassignment and APPLCTN macro 58

E

- ETO
 - descriptors
 - creating, rules for 400
 - format 399
 - logon 401
 - MFS device 405
 - MSC 406
 - overrides 401
 - user 406
 - IMSCTRL macro, ETOFEAT 87, 399
 - including in IMS 23, 399
 - stage 1 48
 - terminal naming rules 54
- examples
 - coding conventions 56
 - Data Communication macro statements 434
 - data sharing system configuration 455
 - IMS DB/DC system definition 427
 - IMS DBCTL environment 458
 - MSC 450
- execution parameters
 - categories and purpose 375
 - data communication 367, 372, 376
 - database 365
 - db buffer sizes 365
 - db performance options 366
 - dc buffer sizes 367
 - dc performance options 367
 - performance-related 373
 - PSB-related 375
 - recovery-related 370, 374, 377
 - region control 372, 376
 - security-related 370, 374, 378
 - specifying 365
 - system
 - active regions 368
 - DL/I address space 368
 - IRLM options 369
 - MVS options 368
 - nucleus identifier 367
 - performance options 368

F

Fast Path

- DB2, and 419
- DBF trace entries 353
- DEDB buffers pools, defining 340
- IMSFP procedure 236
- macro statements
 - APPLCTN 58, 59
 - DATABASE 75
 - FPCTRL 76
- terminals
 - 2740 terminal 140

FDBR

- FDR procedure 217
- including in IMS 25

FDR procedure 217

Finance work station 142

fixed pages, defining 309

FPCTRL

- description 76
- parameters 77
- syntax diagram 76

FPUTIL procedure

- description 221
- example 221

FRE

- controlling 8
- estimating 8
- MFS, and 465
- parameter, BUFPOOLS macro 63

H

hash table slots, specifying 409

HIDAM databases

- declaring 9
- macro instructions 75

high-speed sequential control statements, specifying

- examples 394
- SETO 391
- SETR 393
- syntax 391

high-use program modules, making resident with

- DFSMPLEX 318

I

IDLE command 127

IDLIST macro

- CONFIG macro, and 73
- description 78
- parameters 79
- syntax diagram 78

IMS procedure

- description 221
- MVS, and 382
- storing 191

IMS.PROCLIB members

- altering 300
- DB2, and 419
- DBFMSDBx 304

IMS.PROCLIB members (continued)

- DFS62DTx 363
- DFSDCxxx 305
- DFSDFRxx 307
- DFSFIXnn 309
- DFSHSBxx 313
- DFSINTxx 318
- DFSPBDBC 320
- DFSPBDCC 320
- DFSPBIMS 321
- DFRSRxx 321
- DFSSPMxx 332
- DFSSQxxx 336
- DFSVFLqq 338
- DFSVSMxx 339
- execution, preparing for 303
- generated 299
- initializing 301
- modifications, controlling 303
- renaming 300
- specifying 304
- tailoring 299, 301

IMS transactions

- defining 12
- TRANSACT macro, using 12

IMSBATCH procedure 232

IMSCOBGO procedure 233

IMSCOBOL procedure 235

IMSCTF macro

- description 79
- parameters 80
- syntax diagram 79

IMSCTRL macro

- description 82
- parameters 84
- syntax diagram 82
- system definition types 29

IMSFP procedure 236

IMSGEN macro

- assembler and linkage editor options 99
- COMM macro, and 137
- description 95
- general communication options 106
- IMS data set options 102
- JCL statements 104
- MFS, and 461
- parameters 99
- sample IMSGEN macro statement 107
- SECURITY macro, and 137
- security options 107
- syntax diagram 96

IMSMSG job 237

IMSPLI procedure 238

IMSPLIGO 239

IMSRDR procedure

- description 241
- MVS, and 382
- storing 191

IRLM

- CICS, and 415
- coupling facility structure names 360

IRLM (*continued*)
 dumping 293
 DXRJPROC 216
 execution parameters 369
 IMSCTRL macro
 IRLM parameter 88
 IRLMNM parameter 89
 SDUMP 293
 starting 216

L

LGDK device 81
 LGEN
 configuration 30, 43
 preprocessor 42
 SMP/E JCLIN 50
 stage 1 processing 40
 stage 2 processing 41
 storage requirements 45, 46
 system definition 40
 system generation 46
 LINE macro
 CONFIG macro, and 73
 description 108
 parameters 109
 syntax diagram 108
 LINEGRP macro
 CONFIG macro, and 73
 description 112
 parameters 113
 syntax 112
 link-edit order statements 48
 lock manager, selecting
 IRLM 4
 program isolation 4
 LSO (local storage option)
 DL/I address space, specifying 5
 PSB pools, defining 6
 storage considerations 5
 LTERM
 POOL macro 132
 specifying 127
 LU 6 terminal
 TERMINAL macro statement, specifying 140
 VTAM devices 139

M

macro statements
 APPLCTN
 description 56
 online programs, declaring 10
 BUFPOOLS
 description 61
 MFS pool size, controlling 8
 coding conventions 56
 COMM 64
 CONFIG 73
 CTLUNIT 74

macro statements (*continued*)
 DATABASE
 data sharing 9
 online databases, declaring 9
 FPCTRL 76
 guidelines 54
 IDLIST 78
 IMSCTF 79
 IMSCTRL
 checkpoint frequency, setting 3
 data sharing 9
 description 82
 LGEN system definition 40
 MSGEN 95
 LINE 108
 LINEGRP 112
 maximum occurrences 55
 MSGQUEUE 116
 MSLINK 120
 MSNAME 122
 MSPLINK 123
 NAME 127
 POOL 132
 resource naming rules 54
 RTCODE 132
 SECURITY 133
 SPAREA 137
 STATION 137
 SUBPOOL 139
 system configuration macros, use of 2
 system definition macros, use of 1
 TERMINAL
 2740 terminal 140
 2741 terminal 140
 2780 terminal 140
 3270 terminal 140
 3600 work station 140
 Finance work station 140
 LU 6 140
 SLU 1 140
 SLU 2 140
 SLU P 140
 SPOOL 140
 System/3 terminal 140
 System/7 terminal 140
 TRANSACT
 description 173
 IMS transactions, defining 12
 TYPE 186
 VTAMPOOL 188
 macro table, using 31
 master terminal
 configuring 22
 devices, choosing 21
 specifying 21, 127, 130
 message format pool
 size, estimating 8
 size, specifying 367
 Message Processing Region
 data communication EXEC parameters 372, 376
 parameters, categories and purposes 371

Message Processing Region (*continued*)
performance-related parameters 373
PSB-related EXEC parameters 372, 375
recovery-related parameters 370, 374, 377
region-control EXEC parameters 372, 376
security-related EXEC parameters 370, 374, 378

MFS

buffer pool use 465
BUFPOOLS macro 461
COMM macro 461
CONFIG macro 464
FREs 465
IMSGEN macro 461
INDEX directory 464
MSGQUEUE macro 461
pool space, allocating 8
pool space, defining 8
system definition and programming
considerations 461
TERMINAL 462
TYPE macro 462

MPPs and DB2 419

MSC

APPLCTN macro, and 58
example 450
macro statements, preparing 18

MSDB

DBFMSDBx member 304

MSGQUEUE macro

description 116
MFS, and 461
parameters 116
syntax diagram 116

MSLINK macro

description 120
parameters 121
syntax diagram 120

MSNAME macro

description 122
parameters 123
syntax diagram 122

MSPLINK macro

description 123
parameters 124
syntax diagram 123

MVS

IMS procedures, and 382
IMSRDR procedures, and 382

N

NAME macro

description 127
parameters 129
syntax diagram 128

O

ODBA

accessing IMS databases 411
overview 411

ODBA (*continued*)

setup 411
online databases, declaring 9
online programs
characteristics 10
declaring 10
Open Database Access (ODBA) interface 411
order statements, link-edit 48
OS/390 application programs
accessing IMS databases using ODBA 411
OSAM
buffer pools, defining 347, 378, 379
coupling facility 361, 362
I/O errors, tracing 353
sequential buffering 381
subpools, defining 347, 381
OTMA
control flow, tracing 355
DCC procedure 201
generic resource (GRNAME) parameter 257
message prefix sizes 118

P

performance-related EXEC parameters 373
PL/I modules, organizing 382
PLITDLI 243
POOL macro 132
preinitialization routines, specifying in DFSHSBxx 318
preprocessor
executing 43
parameters, defining 44
storage requirements, estimating 45
procedures
DBBBATCH 192
DBC 195
DBRC 200
DCC 201
DD statements 279
DFSMPR 211
DLIBATCH 212
DLISAS 214
DXRJPROC 216
FDR 217
FPUTIL 221
IMS 221
IMSBATCH 232
IMSCOBGO 233
IMSCOBOL 235
IMSFP 236
IMSPLI 238
IMSPLIGO 239
IMSRDR 241
parameters 244
RDIBATCH 242
storing 191
sysgen-supplied, cataloged 189, 191
program modules, making high-use resident 318
programs
application, characteristics of 10
online, declaring 10

- PSB performance options
 - choosing 11
 - dynamic 11
 - not resident 11
 - resident 11
- PSB pools, defining
 - /DISPLAY POOL PSBP command 6
 - ACBGEN utility 6
 - BUFPOOLS macro 61
 - performance options 11
- PSB-related EXEC parameters 372, 375

R

- RACF
 - DLISAS procedure 215
 - IMS procedure 221
- RDIBATCH procedure 242
- recovery-related EXEC parameters 370, 374, 377
- region-control EXEC parameters 372, 376
- regions, dependent
 - dumping options 291
 - number of, choosing the 3, 90, 269
 - preinitialization routines 318
- resource naming rules
 - ETO terminals 54
 - macros 54
- restarting IMS
 - checkpoints, using 4
- RMF
 - accounting procedures 7
- RSR
 - COMM macro
 - APPLID parameter 66
 - PSWD parameter 70
 - DFSRSRxx member 321
 - DL/I batch jobs, 193
 - IMSCTRL macro, RSRFEAT parameter
 - GSGNAME parameter 88
 - RSRFEAT parameter 91
 - TMINAME parameter 95
 - including in IMS 25
 - log router activity, tracing 354
 - options, specifying 321
- RTCODE macro
 - description 132
 - parameters 132
 - syntax diagram 132

S

- SBCO 387
- SBESNAP 389
- SBIC 387
- SBONLINE control statement 350
- SBPARAM 384
- SBSNAP 388
- security
 - EXEC parameters 370, 374, 378
 - options 7
- SECURITY macro
 - COMM macro, and 137

- SECURITY macro (*continued*)
 - description 133
 - IMSGEN macro, and 137
 - parameters 134
 - syntax diagram 133
- sequential buffering
 - CICS, and 417
 - control statements, specifying
 - SBCO 387
 - SBESNAP 389
 - SBIC 387
 - SBPARAM 384
 - SBSNAP 388
 - SNAPDEST 390
 - online system 350
 - OSAM 378, 381
- serviceability and trace options, defining 350
- SETO 391
- SETR 393
- shared queues
 - interface, tracing 355
- SLU 1
 - TERMINAL macro statement, specifying 140
- SLU 2
 - TERMINAL macro statement, specifying 140
- SLU P
 - TERMINAL macro statement, specifying 140
- SMF
 - accounting procedures 7
- SMP/E
 - LGEM definition 50
 - maintaining stage 1 output 50
- SNAPDEST 390
- Sort/Split utility
 - LGEM Stage 1 processing 41
- SPAREA macro 137
- Spool
 - TERMINAL macro statement, specifying 140
- stage 1
 - ETO option 48
 - IMS.PROCLIB updates 299
 - input, sequencing 40
 - input statements, entering 47
 - LGEM processing 40
 - link-edit order statements 48
 - maintaining output with SMP/E 50
 - output 48
 - system definition, running 47
- stage 2
 - input 48
 - output 49
 - running 49
- START command
 - database access, defining 75
 - DATABASE macro, ACCESS parameter 75
 - dependent regions, starting 3
 - external subsystems, starting 421, 422
 - procedure, naming a 303
- STATION macro 137
 - description 137
 - parameters 138

STATION macro 137 (*continued*)
 syntax diagram 137

storage estimates
 extended private storage 45, 46

storage pool definitions
 default 332
 DFSSPMxx, overriding defaults 332

SUBPOOL macro
 description 139
 parameters 140
 syntax diagram 140

subsystem identification parameters 369

System/3
 LINEGRP macro statement, specifying 112
 TERMINAL macro statement, specifying 140

System/7
 LINEGRP macro statement, specifying 112
 TERMINAL macro statement, specifying 140

system definition
 ALL 29
 BATCH 29
 classes
 DB/DC 29
 DBCTL 29
 DCCTL 29
 CTLBLKS 29
 examples
 DB/DC environment 427
 DC macro statements 434
 IMSCTRL macro 82
 link-edit order statements 48
 managing 40
 MODBLKS 29
 MSVERIFY 29
 NUCLEUS 29
 ON-LINE 29
 preprocessor
 description 42
 executing 43
 LGEN 45
 parameters, defining 44
 standard 45
 storage requirements, estimating 45
 stage 1, running 47
 stage 1 input 47
 stage 1 output 48
 stage 2, running 49
 stage 2 input 48
 stage 2 output 49
 verifying 41

system generation
 storage requirements 46

T

terminal devices, allocating in an online system 192

TERMINAL macro
 COMM macro, and 72
 description 140
 MFS 462
 parameters 153

TERMINAL macro (*continued*)
 syntax diagram 142

TRACE command
 DASD log activity, tracing 353
 DB2 subsystem connection, tracing 356
 DBF entries from FP, tracing 353
 dispatcher activity, tracing 352
 DL/I activity, tracing 353
 latch activity, tracing 354
 lock activity, tracing 354
 options 352
 OTMA control, tracing 355
 queue manager, tracing 355
 RSR log router, tracing 354
 scheduler, tracing 355
 shared queues interface, tracing 355
 storage manager calls, tracing 355

trace entries
 options, defining 350
 storage requirements 356

TRANSACT macro
 description 173
 parameters 176
 syntax diagram 174

transport manager subsystem
 installing 295
 VTAM definition 295

tuning considerations
 control program address spaces 7

TYPE macro
 description 186
 MFS 462
 parameters 187
 syntax diagram 186

V

virtual fetch
 DFSVFLqq 338
 using 339

VSAM
 buffer pools, defining 341, 378, 379
 performance options, defining 346
 subpools, defining 342, 381

VTAM
 CICS, and 415
 COMM macro, and 64, 66, 70, 71
 transport manager system definition 295

VTAMPOOL macro 188

X

XRF
 COMM macro
 APPLID parameter 66
 PASSWD parameter 70
 CTLUNIT macro, BACKUP parameter 74
 DFSHSBxx PROCLIB member 313
 system definition 40

Readers' Comments — We'd Like to Hear from You

IMS/ESA
Installation Volume 2: System Definition and Tailoring
Version 6

Publication No. GC26-8737-05

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Complete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to find	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to understand	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Well organized	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Applicable to your tasks	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? Yes No

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.

Name

Address

Company or Organization

Phone No.



Fold and Tape

Please do not staple

Fold and Tape



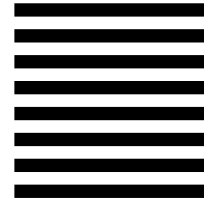
NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Department BWE/H3
P.O. Box 49023
San Jose, CA
95161-9945



Fold and Tape

Please do not staple

Fold and Tape



Program Number: 5655-158



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

GC26-8737-05



Spine information:



IMS/ESA

IMS/ESA V6 Install Volume 2

Version 6