

MAINVIEW[®]
SYSPROG Services
User Guide and Reference

TD-96J-5

Version 3.2

September 30, 1999



This document is published by the BMC Software, Inc., Technical Publications Department.

Copyright © 1999 BMC Software, Inc. All rights reserved.

Copyright © 1997-1999 Boole & Babbage, Inc. All rights reserved.

BMC Software, the BMC Software logo, and all other product or service names are trademarks or registered trademarks of BMC Software, Inc., in the USA and in other select countries. The ™ and ® indicate USA trademark or USA registration.

All other third-party logos and product/trade names are trademarks or registered trademarks of their respective companies.

RESTRICTED RIGHTS LEGEND

Use, distribution, or disclosure by the Government is subject to restrictions set forth in subparagraph (c) (1) (ii) of Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

BMC Software
2101 CityWest Blvd.
Houston, TX 77042-2827
USA

Contacting BMC Software

World Wide Web Address

For a complete list of all BMC Software international offices and subsidiaries, see the BMC Software home page at:

<http://www.bmc.com>

USA and Canada

In the USA and Canada, you can contact BMC Software by telephone, fax, or email:

Telephone	713 918 8800 (Corporate Headquarters) 800 538 1872 (Customer Support)
Fax	713 918 8000 (Corporate Headquarters) 408 526 3897 (Customer Support)
Email	support@boole.com (Customer Support)

International

Outside the USA and Canada, you can contact a BMC Software international support center.

Asia/Pacific/Australia

BMC Software (Australia) Pty. Ltd.
3rd Floor
290 Burwood Road
Hawthorn Victoria 3122
Australia
Telephone: (61) 3 9810 2010
Customer Support: (61) 3 9810 2020
Fax: (61) 3 9810 2030
Email: asiapacific_support@bmc.com

Europe, the Middle East, Africa, and Latin America

BMC Software, Inc.
2101 CityWest Blvd.
Houston, TX 77042-2827
USA
Telephone: 713 918 8800
Fax: 713 918 1302
Email: support@boole.com

You can also call the BMC Software office in your country, if applicable.

Contents

Conventions Used in This Manual viii

Part 1. Introduction 1

Chapter 1. Introduction to SYSPROG Services 3
 SYSPROG Services Features 3
 How SYSPROG Services Operates 4
 Services. 4
 Synchronous Services 4
 Exception Monitor Samplers. 5
 How SYSPROG Services Works with Other Products 5

Chapter 2. Controlling the SYSPROG Services Environment 7
 BBPARM Library Members. 7
 \$\$INEXCL 8
 \$\$INNWTO 9
 \$\$INSYS0 10
 \$\$SRPJ0BS 15
 \$\$RPTEXT 16
 \$\$XENQ 17
 PWSCPM_{xx} 18

Part 2. SYSPROG Services 19

Chapter 3. Using SYSPROG Services 21
 Accessing SYSPROG Services from the Easy Menus. 22
 Accessing SYSPROG from the OS/390 Performance and Control Panel 22
 Accessing SYSPROG Services from the OS/390 Easy Menu 23
 Accessing Services from the SYSPROG Services Menu Panel. 25
 Performing Functions 26
 About SYSPROG Services Command Syntax 29
 Using Commas as Placeholders for Default Values 30
 Executing SYSPROG Services from an Operator Console 31
 Using the MODIFY (F) Command 31
 Using a CMDID Character 31
 Independent Operations 32
 Executing SYSPROG Services as a Started Task 32
 SYSPROG Services as a TSO Command. 33

Part 3. SYSPROG Services Reference. 35

Chapter 4. Synchronous Services. 39
 ALLOCATE 40

APF	41
ASM	47
ASVT	50
AUTHTSO	51
BBXS	55
CDE	57
CHAP	59
CLEAR	61
COMMAND	63
CONSOLES	64
CPU	65
CSA	68
DEALLOC	70
DEVIATN	71
DONTSWAP	73
DSNAME	74
DUMP	76
DVIEW	80
EDTINFO	85
ENQUEUES	89
EQUATE	92
ESCLASS	94
ESTORAGE	99
EXECUTE	101
EXIT	103
FINDMBR	104
INFO	106
IO	107
IPLDATA	109
LABEL	111
LCPU	112
LLIST	114
LNKLST	118
LOADLPA	124
LOG	127
LPA	130
LWAIT	132
LX	134
MCOMMAND	136
MDEVICE	137
MEMSCAN	139
MIO	144
MLCU	147
MONITOR	150
MOUNT	152
MPATH	153
MSTORAGE	155
MTP	157
NUCLEUS	158
OKSWAP	160
PAGING	162
PARMLIST	163
PIO	164
PPT	165

PRIVATE	169
PROGRESS	175
PRSM	177
REPLIES	181
RESERVES	182
RSM	184
SDUMP	188
SETOP	191
SMFEXIT	192
SMSLIST	201
SOFTFRR	203
SPACE	207
SRM	209
SSVT	212
STATUS	214
SUBMIT	217
SVCFIND	218
SYSDUMP	220
TCB	221
TERM	224
TIME	225
TIOT	226
TPIO	229
TQE	231
TRACK	234
TRTBL	237
TSULIST	240
UCB	242
USING	245
VIO	247
VMCMD	248
WARNING	249
ZAP	251
Chapter 5. Exception Monitor Samplers	253
ACPU	254
AIO	256
AIOR	258
APAG	260
APGR	262
ASRV	264
ASSR	266
BCPU	268
BIO	269
BPAG	271
BSRV	273
CHA	275
CPU	277
CQD	279
CSA	280
CSMJ	282
DEV	284
DIOQ	285
DNR	287
DOM	288

DVTN	289
ENQ	291
FIX	292
INT	293
JET	294
JSU	296
JVR	298
LOG	299
MTP	302
NRQ	303
NRQP	305
OUT	307
PAG	308
PGQ	309
PGSR	310
PSRV	312
REP	314
RES	315
RSU	316
SMF	317
SPA	319
SRM	320
SWP	321
TCPU	322
TIO	324
TPAG	326
TSRV	328
WDLY	330
WOBJ	331
WRT	332
WTO	333
WTOR	334

Part 4. Glossary and Index	335
Glossary	337
Index	347

About This Book

This book describes the services offered by the SYSPROG Services software. It contains information about accessing and invoking SYSPROG Services, and a reference section that provides detailed information about each service.

Who Should Read This Book

This book is intended for system programmers, developers, operations personnel, and applications developers.

How This Book Is Organized

There are four parts to this manual, consisting of five chapters, a glossary, and an index.

[Part 1, “Introduction” on page 1](#), contains the following information:

- [Chapter 1, “Introduction to SYSPROG Services” on page 3](#), provides a general overview of the service functionality of SYSPROG Services.
- [Chapter 2, “Controlling the SYSPROG Services Environment” on page 7](#), describes how to control various aspects of the SYSPROG Services operating environment.

[Part 2, “SYSPROG Services” on page 19](#), contains the following information:

- [Chapter 3, “Using SYSPROG Services” on page 21](#), describes how to access and invoke SYSPROG Services and provides information about interpreting the service output of SYSPROG Services.

[Part 3, “SYSPROG Services Reference” on page 35](#), contains the following information:

- [Chapter 4, “Synchronous Services” on page 39](#), provides detailed information about each service, including command syntax, parameter descriptions, and output examples.
- [Chapter 5, “Exception Monitor Samplers” on page 253](#), describes how the Exception Monitor helps you detect potential problems by monitoring thresholds that you establish, and lists the warning messages that are issued when a threshold is exceeded.

[Part 4, “Glossary and Index” on page 335](#), contains the glossary and index.

Recommended Reading

You will need to refer to the following books to complete the installation and implementation of MainView for OS/390:

- *Product Installation and Maintenance Guide*
- *MAINVIEW SYSPROG Services Customization Guide*
- *MainView for OS/390 System Programmer Services Quick Reference Guide*

- *MainView Alternate Access Implementation and User Guide*
- *MainView for OS/390 User Guide and Reference*

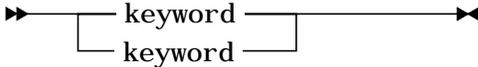
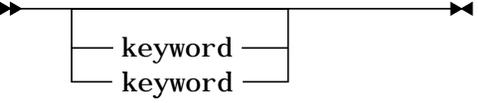
Conventions Used in This Manual

This section documents the syntax conventions used throughout this reference manual and identifies wildcard characters.

Syntax Diagrams

Syntax diagrams are used throughout this reference manual to illustrate syntax usage for services. Syntax diagrams graphically depict the relationship between literal and variable (or other notational) items as understood by an application program or programs. Consult the following table for the meaning of conventions used in these diagrams.

Convention	Meaning
▶▶ (double right arrowheads)	Indicate the beginning of a syntax diagram.
▶ (single right arrowhead)	Can have two meanings: <ul style="list-style-type: none"> • When found at the beginning of a line, it indicates that the syntax diagram is continued from another line or diagram. • When found at the end of a line, it indicates that the syntax diagram continues to another line or diagram.
◀◀ (opposing arrowheads)	Indicate the end of a syntax diagram.
▶▶—keyword—▶▶	An item on the main line of the diagram is required.
▶▶—literal—▶▶	A literal is represented in flat text.
▶▶— <i>variable</i> —▶▶	A variable is represented in italic text.
▶▶ ┌ keyword ─┐ └──────────┘	An item above the main line of the diagram is the default.
▶▶ └── keyword ─┘	An item underneath the main line of the diagram is optional.
▶▶ ┌── keyword ─┐ └──┬────────┘	Anything under the upward left-branching arrow can be repeated.

Convention	Meaning
	A vertical list of options requires you to choose one.
	A vertical list of options with the topmost option left blank indicates that you can choose one option or none.

Part 1. Introduction

This part introduces you to SYSPROG Services and discusses how to control the SYSPROG Services environment.

Chapter 1. Introduction to SYSPROG Services	3
SYSPROG Services Features	3
How SYSPROG Services Operates	4
Services	4
Synchronous Services	4
Exception Monitor Samplers	5
How SYSPROG Services Works with Other Products	5
Chapter 2. Controlling the SYSPROG Services Environment	7
BBPARM Library Members	7
\$\$INEXCL	8
\$\$INNWTO	9
\$\$INSYS0	10
\$\$RPJOBS	15
\$\$RPTEXT	16
\$\$XENQ	17
PWSCPMxx	18

Chapter 1. Introduction to SYSPROG Services

SYSPROG Services is a realtime software tool that helps your data center manage the day-to-day performance and operation of the OS/390 environment by helping you to detect, diagnose, and correct problems as they occur.

The synchronous services and Exception Monitor samplers of SYSPROG Services provide the tools for online system management. Data center personnel can delegate tasks to SYSPROG Services by storing a series of operational commands that can be executed automatically.

This chapter describes:

- The features of SYSPROG Services
- The various operation modes of SYSPROG Services
- The services offered by SYSPROG Services
- The relationship between SYSPROG Services and other BMC Software products

SYSPROG Services Features

The following table lists the features of SYSPROG Services, along with a brief description:

Table 1. SYSPROG Services Features

Feature	Description
Synchronous services	Lets you examine overall system performance and isolate specific system performance delays. Services are invoked when entered.
Exception Monitor samplers	Provide early detection of potential performance problems through user-defined threshold indicators. Highlighted warning messages notify you of impending system problems.
Password protection facility	Prevents unauthorized use of SYSPROG Services.
SAF security interface	Provides an interface to an external security manager (ESM).

You can use the features of SYSPROG Services to increase data center productivity, improve system throughput, and manage job deadlines.

How SYSPROG Services Operates

The SYSPROG Services system operates as an authorized problem program in its own address space (or within a TSO address space). You can run SYSPROG Services in the operation mode that best suits your needs. [Table 2](#) lists each of the available modes and gives a brief description of each.

Table 2. SYSPROG Services Operating Modes

In this mode	You can
PAS	Invoke services from a user address space (UAS) using the MODIFY (F) command.
TSO	Invoke services by issuing a TSO command in line mode. Any TSO user with operator authority can access SYSPROG Services.
Started Task	Invoke services by running SYSPROG Services from the system console or a dedicated terminal as an addition to the operating system environment. You can use the MODIFY (F) command to invoke services.
Batch Job	Submit a list of SYSPROG Services commands for batch processing. This mode is useful if you want to collect and print out data from a series of services.

For more information on the operation modes, refer to the *MAINVIEW SYSPROG Services Customization Guide*.

SYSPROG Services interacts with the operator by using the following facilities:

- Write to Operator with Reply (WTOR)
- MODIFY (F) command entry
- TPUT/TGET sequences for TSO users
- Product Address Space (PAS)

Services

SYSPROG Services provides two kinds of services: synchronous services and Exception Monitor samplers.

Synchronous Services

The synchronous services are invoked as soon as you request them. For example, if you are running SYSPROG Services in TSO mode, you invoke the CPU synchronous service by

entering the command CPU. Parameters are entered at the time you invoke the service. See [“Synchronous Services” on page 39](#).

Exception Monitor Samplers

SYSPROG Services Exception Monitor samplers warn you about resource imbalances and shortages. When an Exception Monitor sampler exceeds its threshold value (which you set in a BBPARM library member), SYSPROG Services issues a warning message, indicating a potential problem. See [Chapter 5, “Exception Monitor Samplers” on page 253](#).

How SYSPROG Services Works with Other Products

SYSPROG Services works in conjunction with other BMC Software products, both as a component and as an instrument for gathering data.

If you are running SYSPROG Services as a component of MainView for OS/390, the services are accessed from the SYSPROG Services menu, see [“Accessing SYSPROG Services from the OS/390 Easy Menu” on page 23](#).

Chapter 2. Controlling the SYSPROG Services Environment

This chapter describes the BBPARM library and how you can use it to control various aspects of the SYSPROG Services operating environment.

BBPARM Library Members

You can control the operations of the services by modifying a partitioned data set (PDS) containing members that set initial values for the services functions. This PDS must be allocated to a DD name in the address space where SYSPROG Services executes; this DD name is //BBPARM. When you initialize SYSPROG Services in Started Task mode or Batch mode, the values stored in BBPARM library members are read and placed in storage, where they remain in effect for each service. If SYSPROG Services cannot find the DD name //BBPARM, default values are used instead.

If you are running SYSPROG Services in TSO mode, you can use BBPARM library members by allocating the BBPARM data set to your TSO address space and giving it the DD name //BBPARM.

Note: For compatibility with previous versions of SYSPROG Services, //LIB is also supported for the DD name //BBPARM.

You also can create BBPARM members containing lists of SYSPROG Services, OS/390, or JES commands and use the EXECUTE service to execute these lists. See [“EXECUTE” on page 101](#) for more information.

[Table 3](#) lists each BBPARM library member along with a brief description and page reference.

Table 3. BBPARM Library Members

Member	Description	See page
\$\$INEXCL	Specifies which job names are excluded from exception reporting by specified Exception Monitor services.	8
\$\$INNWTO	Restricts SYSPROG Services command entry to Modify mode.	9
\$\$INSYS0	Defines runtime parameters for SYSPROG Services.	10
\$\$RPJOBS	Exempts jobs with constantly outstanding WTORs from the SYSPROG Services outstanding reply scan.	15
\$\$RPTEXT	Exempts constantly outstanding WTORs from the SYSPROG Services outstanding reply scan.	16
\$\$XENQ	Specifies major names to be excluded from the ENQUEUEES service contention display and from the Exception Monitor ENQ sampler warning messages.	17
PWSCPMxx	Stores selectable sets of Exception Monitor Samplers.	18

The following sections describe the BBPARM library members, including syntax rules and examples.

\$\$INEXCL

This member contains job names you want to exclude from exception reporting by specified Exception Monitor services.

Syntax Rules

The format is

```
service-name jobname1, . . . jobname7
```

where

service-name Is the name of the Exception Monitor service that is to exclude selected job names from exception reporting.

The service names that you can specify are: CPU, ENQ, JET, JVR, JSU, and OUT.

jobname1-jobname7 Are the names of up to seven jobs that are to be excluded from exception reporting.

Defaults

None.

Example 1

To exclude jobs CPUJOB1, CPUJOB2, and CPUJOB3 from exception reporting by the Exception Monitor CPU service, place the following control statement in member \$\$INEXCL:

```
CPU CPUJOB1, CPUJOB2, CPUJOB3
```

Example 2

To eliminate TCAS and DUMPSRV address spaces from exception reporting by the OUT service (which monitors swapped-out address spaces), place the following control statement in member \$\$INEXCL:

```
OUT TCAS, DUMPSRV
```

\$\$INNWTO

This member suppresses the WTOR message `AMT000A ENTER SYSPROG Services COMMAND` issued when SYSPROG Services is running as a batch job or started task.

When you run SYSPROG Services as a batch job or started task, you can enter SYSPROG Services commands in two ways:

- By using the OS/390 system command `MODIFY`
- By replying to the WTOR message `AMT000A ENTER SYSPROG Services COMMAND`

If you want to use only the `MODIFY` command to enter SYSPROG Services commands, you can suppress `AMT000A ENTER SYSPROG Services COMMAND` WTOR messages with member `$$INNWTO`. To do this, place member `$$INNWTO` in the `BBPARM` data set; no records need be present.

In addition, you can suppress the `AMT000A ENTER SYSPROG Services COMMAND` WTOR message by specifying an OS/390 subsystem command ID character to identify SYSPROG Services commands. See the description of the `CMDID` keyword in member `$$INSYS0` for more information.

\$\$INSYS0

This member defines run-time parameters for SYSPROG Services. SYSPROG Services reads this member when it initializes.

Syntax Rules

Table 4 lists the keyword parameters in member \$\$INSYS0. The following syntax rules apply:

- Keywords must be entered in the format *keyword=value* (for example, `MSGID=NO`).
- Statements can be continued to the next line if they end with a comma.
- Comment lines start with an asterisk.
- Leading blanks prior to the first keyword are allowed.
- Embedded blanks are not allowed; they terminate the scan of the statements.

Table 4. Keywords for Member \$\$INSYS0

Keyword=Value (default is underlined)	Description
AEWNOROLL= <u>0</u> <i>n</i>	Specifies the number of Exception Monitor sampler messages not to roll. 0 Lets all messages roll; the default. <i>n</i> Lets <i>n</i> messages roll, where <i>n</i> is a decimal number from 1 to 10.
AEWROLL= <u>YES</u> NO	Defines whether Exception Monitor sampler messages are allowed to roll off the SYSPROG Services console. This keyword is valid in Started Task and Batch modes only. YES Allows Exception Monitor sampler messages to roll off the SYSPROG Services console; the default. NO Does not allow Exception Monitor sampler messages to roll off the SYSPROG Services console.
AEWSTART= <u>00</u> <i>xx</i>	Defines the suffix for the default PWSCPM _{<i>xx</i>} member containing control statements for Exception Monitor samplers. 00 Uses Exception Monitor control statements in member PWSCPM00 in the BBPARM library; the default. <i>xx</i> Uses Exception Monitor control statements in member PWSCPM _{<i>xx</i>} in the BBPARM library.
Note:	Route codes are only used when sending unsolicited messages (that is, when the messages are not in response to a request or command).

Table 4. Keywords for Member \$\$INSYS0 (Continued)

Keyword=Value (default is underlined)	Description
AEWWTODESCCDE= <u>X' 4000'</u> X' hhhh'	<p>Defines the WTO descriptor code for Exception Monitor sampler messages.</p> <p>X' 4000' Sends WTO messages to descriptor code 2; the default.</p> <p>X' hhhh' Is a hexadecimal bit string that corresponds to the ordinal position of route codes 1 through 16.</p>
AEWWTOROUTCDE= <u>X' 4020'</u> X' hhhh'	<p>Defines the WTO routing code for Exception Monitor sampler messages.</p> <p>X' 4020' Sends WTO messages to routing codes 2 and 11; the default.</p> <p>X' hhhh' Is a hexadecimal bit string that corresponds to the ordinal position of route codes 1 through 16.</p>
CMDID= <i>char</i>	<p>Defines an OS/390 subsystem command character. This keyword is valid in Started Task and Batch modes only.</p> <p><i>char</i> Is a command identification character to route commands to SYSPROG Services using the OS/390 subsystem interface; valid command identification characters are:</p> <p style="text-align: center;"> @ # % - & * () _ - + = " . < > / ? ¢</p>
EOSMSG= <u>NO</u> YES	<p>Specifies whether SYSPROG Services issues an END OF SERVICE message at the completion of every service. This keyword is valid in Started Task, TSO, and Batch modes only.</p> <p>NO Does not issue an END OF SERVICE message at the completion of every service; the default.</p> <p>YES Issues an END OF SERVICE message at the completion of every service.</p>
EXECLOOP= <u>NO</u> YES	<p>Controls whether the EXECUTE service pauses between issuing commands.</p> <p>NO Causes the EXECUTE service to issue commands without a pause between commands; the default.</p> <p>YES Causes the EXECUTE service to issue commands with a pause between commands.</p>
FNQSIZE= <u>12</u> <i>n</i>	<p>Defines the number of 1K blocks of CSA storage used by Exception Monitor samplers.</p> <p>12 Specifies twelve 1K blocks of CSA storage; the default.</p> <p><i>n</i> Is a decimal number of 1K blocks of CSA storage.</p>
<p>Note: Route codes are only used when sending unsolicited messages (that is, when the messages are not in response to a request or command).</p>	

Table 4. Keywords for Member \$\$INSYS0 (Continued)

Keyword=Value (default is underlined)	Description
JES2TPUT= <u>NO</u> YES	<p>Specifies whether JES2 redirects TPUTs.</p> <p>NO Does not redirect JES2 TPUTs; the default.</p> <p>YES Redirects JES2 TPUTs.</p>
LCLAEW= <u>CONSOLE</u> TPMON BOTH	<p>Defines where Exception Monitor sampler messages are sent.</p> <p>CONSOLE Sends Exception Monitor sampler messages to the OS/390 console; the default.</p> <p>TPMON Sends Exception Monitor sampler messages to the LCL3270 device.</p> <p>BOTH Sends Exception Monitor sampler messages to both the OS/390 console and the LCL3270 device.</p>
LCLAEWTS= <u>NO</u> YES	<p>Specifies whether Exception Monitor sampler messages have a timestamp.</p> <p>NO Does not timestamp Exception Monitor sampler messages; the default.</p> <p>YES Timestamps Exception Monitor sampler messages.</p>
LLA=(yy,zz)	<p>Specifies the suffixes of the CSVLLA members used to remove and reinstate the link list data set concatenation from control of the LLA address space. The CSVLLA members reside in the library pointed to by IEFPARM in the LLA address space (by default, SYS1.PARMLIB). This keyword is used in conjunction with the LLIST service.</p> <p>yy Uses member CSVLLAyy to specify the removal of the link list data set concatenation from control of the LLA address space; used if you omit yy on the LLIST command.</p> <p>zz Uses member CSVLLAzz to place the link list data set concatenation back under control of the LLA address space; used if you omit zz on the LLIST command.</p>
MAXSERVS= <u>3</u> <i>n</i>	<p>Specifies the maximum number of concurrent services in SYSPROG Services. This keyword is valid in Started Task and Batch modes only.</p> <p>3 Allows up to three concurrent services; the default.</p> <p><i>n</i> Allows up to <i>n</i> concurrent services.</p>
<p>Note: Route codes are only used when sending unsolicited messages (that is, when the messages are not in response to a request or command).</p>	

Table 4. Keywords for Member \$\$INSYS0 (Continued)

Keyword=Value (default is underlined)	Description
MSGID= <u>YES</u> NO	<p>Specifies whether message identification numbers prefix SYSPROG Services messages. This keyword is not valid for TSO mode.</p> <p>YES Message identification numbers prefix all messages; the default.</p> <p>NO No message identification numbers prefix any messages.</p> <p>Note: Message identification numbers appear in tracking output even if you specify <code>MSGID=NO</code>.</p>
NOROLL= <u>0</u> <i>n</i>	<p>Specifies the number of Exception Monitor sampler messages not to roll.</p> <p>0 Lets all messages roll; the default.</p> <p><i>n</i> Lets <i>n</i> messages roll, where <i>n</i> is a decimal number from 1 to 10.</p>
OPRAUTH= <u>NO</u> YES	<p>Defines the need for TSO operator authority for SYSPROG Services that may require TSO operator authority. This keyword is valid for TSO mode only.</p> <p>NO TSO users must have operator authority; the default.</p> <p>YES TSO users do not need operator authority.</p>
TGETINT= <u>1</u> <i>n</i>	<p>Specifies the interval between TGETs. This keyword is valid for TSO mode only.</p> <p>1 Issues a TGET every 1 minute; the default.</p> <p><i>n</i> Issues a TGET every <i>n</i> minutes.</p>
TPUTINT= <u>1</u> <i>n</i>	<p>Specifies the interval between TPUTs. This keyword is valid for TSO mode only.</p> <p>1 Issues a TPUT every 1 minute; the default.</p> <p><i>n</i> Issues a TPUT every <i>n</i> minutes.</p>
TTFSIZE= <u>0</u> <i>n</i>	<p>Defines the size of the SYSPROG Services trace table.</p> <p>0 There is no trace table; the default.</p> <p><i>n</i> Allocates a trace table with <i>n</i> entries.</p>
WQEDISP= <u>0</u> <i>n</i>	<p>Defines displacement into locally modified WQEs.</p> <p>0 Displacement is zero for standard shops; the default.</p> <p><i>n</i> Indicates a displacement into the WQE.</p>
<p>Note: Route codes are only used when sending unsolicited messages (that is, when the messages are not in response to a request or command).</p>	

Table 4. Keywords for Member \$\$INSYS0 (Continued)

Keyword=Value (default is underlined)	Description
WTODESCCDE= <u>X' 0000'</u> X' hhhh'	Defines the WTO descriptor code for SYSPROG Services messages. X' 0000' Sends WTO messages to the system log; the default. X' hhhh' Is a hexadecimal bit string that corresponds to the ordinal position of route codes 1 through 16.
WTORDESCCDE= <u>X' 0000'</u> X' hhhh'	Defines the WTOR descriptor code for SYSPROG Services messages. X' 0000' Sends WTOR messages to the system log; the default. X' hhhh' Is a hexadecimal bit string that corresponds to the ordinal position of route codes 1 through 16.
WTOROUTCDE= <u>X' 4060'</u> X' hhhh'	Defines the WTO routing code for SYSPROG Services messages. X' 4020' Sends WTO messages to routing codes 2 and 11; the default. X' hhhh' Is a hexadecimal bit string that corresponds to the ordinal position of route codes 1 through 16.
WTORROUTCDE= <u>X' 4060'</u> X' hhhh'	Defines the WTOR routing code for SYSPROG Services messages. X' 4020' Sends WTO messages to routing codes 2 and 11; the default. X' hhhh' Is a hexadecimal bit string that corresponds to the ordinal position of route codes 1 through 16.
Note: Route codes are only used when sending unsolicited messages (that is, when the messages are not in response to a request or command).	

Example

To prevent SYSPROG Services from prefixing messages with identification numbers, place the following control statement in member \$\$INSYS0:

MSGID=NO

\$\$RPJOBS

This member exempts jobs with constantly outstanding WTORs from the SYSPROG Services outstanding reply scan.

Syntax Rules

The following syntax rules apply:

- Specify one job name per entry.
- Start each entry in column 1.
- Do not place quotation marks around the job name.

Default

None.

Example

Assume jobs IMSDC and TCAM01 have shutdown messages outstanding at all times. You can exclude these jobs from the outstanding reply scan by entering the following statements in member \$\$RPJOBS:

```
IMSDC  
TCAM01
```

\$\$RPTEXT

This member exempts constantly outstanding WTORs from the SYSPROG Services outstanding reply scan.

Syntax Rules

The following syntax rules apply:

- Specify one entry per statement.
- Start each entry in column 1.
- Do not place quotation marks around the message text.
- Enter no more than 16 bytes of message text.

Default

None.

Example

Assume there are 10 versions of a monitor used for terminal control. Each monitor version has its own job name; each job name produces the same message, **ENTER COMMAND FOR RESPONSE**. You can exclude these jobs from the outstanding reply scan by entering the following statement in member **\$\$RPTEXT**:

ENTER COMMAND FO

\$\$XENQ

This member specifies major names and minor names to be excluded from the ENQUEUEES service contention display and from the Exception Monitor ENQ sampler warning messages.

Syntax Rules

The following syntax rules apply:

- List each major name and optional minor name as a single statement.
- Separate the major name and optional minor name with a comma.
- Enter as many statements as necessary.

Default

None.

Example 1

To exclude the major names SYSDSN and SPFEDIT from the ENQUEUEES service contention display and from the Exception Monitor ENQ sampler warning messages, place the following statements in member \$\$XENQ:

```
SYSDSN  
SPFEDIT
```

Example 2

To prevent the ENQUEUEES asynchronous service and the Exception Monitor ENQ sampler from reporting SYSDSN enqueues that begin with SYS1. , place the following statement in member \$\$XENQ:

```
SYSDSN, SYS1.
```

PWSCPMxx

This member stores selectable sets of Exception Monitor services as members in the BBPARM library data set.

Syntax Rules

The member names have the form PWSCPMxx, where xx is alphanumeric (for example, PWSCPMFF, PWSCPM09, PWSCPMA9). Each member contains control statements specifying Exception Monitor services, their warning intervals, and appropriate threshold specifications.

The following syntax rules apply:

- Separate the Exception Monitor service name from the specifications with a comma or at least one blank.
- Enter comments by placing an asterisk in column 1.

Default

By default, the Exception Monitor listed in member PWSCPM00 is invoked when you begin Exception Monitor sampling.

Example

The following is an example of a member called PWSCPMEX:

```
INT 60
ENQ 5, 10
PAG 4, 50
AI OR 4, *, 25, A
```

The example contains four Exception Monitor services:

- The first service (INT 60) specifies an interval rate of 60 seconds.
- The second service (ENQ 5, 10) requests that warning messages be issued at 5-minute intervals if any enqueue conflicts have existed for over 10 minutes.
- The third service (PAG 4, 50) requests that warning messages be issued at 4-minute intervals if the paging rate exceeds 50 over the preceding 4 minutes.
- The fourth service (AI OR 4, *, 25, A) causes warning messages to be issued at 2-minute intervals for each address space whose I/O rate exceeds 25 EXCPs per second over the preceding 2 minutes.

Note: For a description of each Exception Monitor service, refer to the *MainView for OS/390 User Guide and Reference*, TD-E2J.

Part 2. SYSPROG Services

This part shows you how to access and invoke services.

Chapter 3. Using SYSPROG Services	21
Accessing SYSPROG Services from the OS/390 Easy Menu	22
Accessing Services from the SYSPROG Services Menu Panel	24
Performing Functions	25
About SYSPROG Services Command Syntax	28
Using Commas as Placeholders for Default Values	29
Executing SYSPROG Services from an Operator Console	30
Using the MODIFY (F) Command	30
Using a CMDID Character	30
Independent Operations	31
Executing SYSPROG Services as a Started Task	31
SYSPROG Services as a TSO Command	32

Chapter 3. Using SYSPROG Services

SYSPROG Services provides you with the means to control certain internal aspects of your OS/390 systems. For example, you can display and alter storage, list the DD names used by a job, add modules and SVCs to the Link Pack Area, and perform a variety of other OS/390 system tasks.

If you are running SYSPROG Services through MainView for OS/390, you can access the services in the following ways:

- From the OS/390 Performance and Control menu, select **SYSPROG**.
- From the SYSPROG Services Easy Menu, EZMSPROG, select **SYSPROG Fast Menu**, for an expanded set of options.
- From the OS/390 Easy Menu, under the **Utilities** category, select **SYSPROG Services**.
- In TSO line mode, on the **COMMAND** line, type **SYSPROG** .
- In Started Task mode, use the MODIFY (F) command to invoke a specific service from an operator console.

Accessing SYSPROG Services from the Easy Menus

You can access SYSPROG Services from the Easy Menus in different ways. If you are a MainView for OS/390 customer, you can select **MV390**, or, to access SYSPROG Services directly, select **SYSPROG**. Below is an example of the OS/390 Performance and Control Panel

```

----- OS/390 Performance and Control -----
Option  ===>

MainView Online Applications
  1  MV390   Execute MainView for OS/390
  2  CMF    Execute CMF MONITOR Online
  3  SYSPROG Execute MainView for SYSPROG Services

Additional Online Components
  4  CSMON   Common Storage Monitor
  5  CMFMON  Invoke CMFMON realtime analysis
  6  CMFUTIL CMF Extractor Online Utilities

Batch Applications
  7  ANALYZER Generate CMF Analyzer batch reports
  8  MVBATCH  Generate MainView for OS/390 batch reports

General Services
  M  MESSAGES Display Messages and Codes
  N  NOTES    Release Notes

                                     Copyright BMC Software, Inc. 1999

```

Figure 1. OS/390 Performance and Control Panel

When you select MV390 from the OS/390 Performance and Control panel, the OS/390 Easy Menu panel (EZM390) is displayed. See [“Accessing SYSPROG Services from the OS/390 Easy Menu” on page 23](#).

Accessing SYSPROG from the OS/390 Performance and Control Panel

When you select SYSPROG from the OS/390 Performance and Control panel, the SYSPROG Easy Menu panel (EZMSPROG) is displayed.

To help you access the service you need, options on the SYSPROG Easy menu are grouped according to their area of functionality and have been given intuitive, descriptive names. [Figure 2](#) shows an example.

```

09DEC1998  10: 37: 03  ----- INFORMATION DISPLAY -----
COMMAND  ==>
CURR WIN  ==> 1          ALT WIN  ==>
WI  =EZMSPROG=====SYSE=====*=====09DEC1998==10: 37: 02====MVMVS=====1
                                SYSPROG Easy Menu

  Job Info                      System Info
> Actions                       +-----+ > Actions
> Performance                   | Place cursor on | > Dump Services
> Storage                       | menu item and  | > Performance
                                | press ENTER    | > Storage
                                +-----+ > Utilities

  Device Info                   Advanced
> I/O Subsystem                 > SYSPROG Fast Menu
> Realtime Performance
> Utilities                      . Return...
    
```

Figure 2. SYSPROG Services Menu (EZMSPROG)

Accessing SYSPROG Services from the OS/390 Easy Menu

You can access SYSPROG Services from the OS/390 Easy Menu (EZM390) by selecting the SYSPROG Services option under the Utilities category, as shown in [Figure 3](#), or by typing **EZMSPROG** on the **COMMAND** line. The primary SYSPROG Services menu is called the SYSPROG Easy Menu (EZMSPROG).

```

15MAR1999  14: 36: 32  ----- INFORMATION DISPLAY -----
COMMAND  ==>
CURR WIN  ==> 1          ALT WIN  ==>
WI  =EZM390=====DXTSTJ=====*=====15MAR1999==14: 36: 31====MVMVS=====1
                                OS/390 Easy Menu

  Activity                      Time frame - Interval    Utilities
> System Overview               +-----+ > SYSPROG Services
> Jobs                         | Place cursor on | . Program and I/O Trace
> Devices                       | menu item and  | > Alarm Management
> Data Set Usage                | press ENTER    | > OS/390 Fast Menu
> Storage                       +-----+ > RMF-like Menus
> XCF Monitoring                > Environment Settings
> Coupling Facility              . Return...
> WLM Workloads
> Non-WLM Workloads
    
```

Figure 3. Select SYSPROG Services from the Utilities Category

To help you access the service you need, options on the EZMSPROG menu are grouped according to their area of functionality and have been given intuitive, descriptive names. EZMSPROG is shown in [Figure 4](#).

```

09DEC1998 10:37:03 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
W1 =EZMSPROG=====SYSE=====*=====09DEC1998==10:37:02====MVMVS=====1
                                SYSPROG Easy Menu

  Job Info
> Actions
> Performance
> Storage

  Device Info
> I/O Subsystem
> Realtime Performance
> Utilities

  System Info
> Actions
> Dump Services
> Performance
> Storage
> Utilities

  Advanced
> SYSPROG Fast Menu

. Return...

```

Figure 4. SYSPROG Services Menu (EZMSPROG)

SYSPROG Services Fast Menu

The SYSPROG Services Fast Menu, an option on SYSPROG Easy Menu as shown in [Figure 5 on page 24](#), provides more direct access to the services. You can access it by selecting SYSPROG Fast Menu from SYSPROG Easy Menu, or by typing **EZMFPROG** on the command line. .

```

15MAR1999 14:48:23 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
W1 =EZMSPROG=EZMFPROG=DXTSTJ====*=====15MAR1999==14:48:08====MVMVS=====1
                                SYSPROG Fast Menu

  Job Info
. Change Priority
. Force
. Make Non-Swappable
. Make Swappable
. Master Console Command
. Take a SVC Dump

  Device Info
. Channel Activity
. Device Activity
. EDT Information
. Enqueues
. Mounts Pending
. Outstanding I/Os
. Reserves
. Storage Groups
. UCB Information

  Menu
> Job Performance
> Job Storage
> SYSPROG Easy Menu
> System Performance
> System Storage

  System Info
. Console Buffers
. List Dump Data Sets
. OS/390 Release Le
. Outstanding Replies
. Software Logrecs

  System Actions
. APF
. Clear Dump Data Sets
. IKJTS0xx List/Update
. Linklist List/Update
. Load LPA
. SMFEXIT List/Update
. VM Command
. ZAP Memory
. Return...

```

Figure 5. SYSPROG Services Fast Menu (EZMFPROG)

Accessing Services from the SYSPROG Services Menu Panel

An alphabetical list of all available services is displayed on the SYSPROG Services Menu panel, as shown in [Figure 6](#).

To view this panel from SYSPROG Services, on the **COMMAND** line, type **SYSPROG**.

To view this panel from MainView for OS/390

1. From the OS/390 Easy Menu panel, select **SYSPROG Services**
2. On the **COMMAND** line, type **SYSPROG**.

The SYSPROG Services Menu panel is shown in [Figure 6](#).

```

----- SYSPROG SERVICES MENU ----- Row 1 of 89
COMMAND ==>                               SCROLL ==> PAGE
Valid line commands are:                   Valid COMMANDs are: TARGET - SYSE
S - Select service panel                   L - Locate a service in the list
E - Execute service                         service - Execute a service
                                           MENU == RXAMAIN

Service Parameters  Description
-----
ALLOCATE           MARKS DEVICE AS ALLOCATED AND ONLINE
APF                LIST CURRENT APF DATA SETS AND VOLUMES
ASM               DISPLAY AUXILIARY STORAGE MANAGER DATA
AUGHTSO          DISPLAY/BUILD TSO LISTS FROM IKJTSOXX
BBXS             MANAGE BOOLE AND BABBAGE SUBSYSTEM SERVICES
CDE              LIST INFORMATION ABOUT LOADED MODULES
CHAP             CHANGE ADDRESS SPACE DISPATCHING PRIORITY
CLEAR           CLEAR SYSTEM DUMP DATA SETS
CONSOLES        DISPLAY MCS CONSOLE BUFFER USAGE
CPU             DISPLAY CPU USAGE BY JOB
CSA             DISPLAY SYSTEM USAGE OF CSA
CSMON          DISPLAY COMMON STORAGE USAGE BY ADDRESS SPACE
DEALLOC        MARKS DEVICE AS DEALLOCATED
DEVIATN        DISPLAY CONFIGURATION DEVIATION
    
```

Figure 6. The SYSPROG Services Menu Panel

Scroll down to display the rest of the services. Each service is described in [Part 3, “SYSPROG Services Reference”](#) on page 35.

Performing Functions

You can perform the following actions from the SYSPROG Services Menu panel list:

- Locate a service
- Find out more about a service
- Execute a service

[Table 5](#) summarizes the functions of the SYSPROG Services commands L, S, E, and MODIFY; the subsequent sections provide more details.

Table 3. SYSPROG Services Available commands

To	Do this	For more information
Search for a service by name	Type L <i>service name</i> .	See “ Locating a Service ” on page 26.
Find out more about a service, or change its parameters	Use the S line command.	See “ Finding Out More about a Service ” on page 26.
Execute a service	Do one of the following: <ul style="list-style-type: none"> • Use the E line command (using defaults). • On the COMMAND line, type the name of the service and its parameters. • Use the S line command, specify the parameters on the Service Information panel, and press Enter. 	See “ Executing a Service ” on page 27.
Execute a service from an operator console	Do one of the following: <ul style="list-style-type: none"> • Use the MODIFY command. • Use a predefined CMDID character. 	See “ Executing SYSPROG Services from an Operator Console ” on page 31.

Locating a Service

To locate a service—for example, LNKLST—type the command **L LNKLST**. LNKLST is scrolled to the top of the display.

Finding Out More about a Service

To find out more information about a service, place the cursor under the desired service, type **S** (Select) next to it, and press Enter. The appropriate Service Information panel is displayed, describing the service and providing input fields in which you can specify parameters.

Figure 7 shows the Service Information panel for the DVIEW service.

```

----- SYSPROG - DVIEW Service -----
COMMAND ==>
                                     More:  +
      OPTION ==> 10      (CCHHR, DSN, DSCB,
                        EXTENTS, FILES or LABEL)
      VOLSER ==>        (Volume serial number of
                        DASD device to view)

Required for CCHHR:
      ADDRESS ==>      (Absolute track address,
                        format ccchhhrr)

Required for DSN, DSCB or EXTENTS;
optional partial DName for FILES:
      DSNAME ==>

Optional for DSN:
      SKIP ==>         (Number of blocks to
                        skip--default 0)
      COUNT ==>       (Number of blocks to
                        display--default 1)

Press ENTER to execute service
    
```

Figure 7. Service Information Panel for DVIEW

To execute the service from the Service Information panel, type values in the parameter input fields, if desired, and press Enter. Otherwise, use END to exit the panel.

Executing a Service

To execute a service from the SYSPROG - DVIEW Service panel, do one of the following:

- On the **COMMAND** line, type the name of the service and its accompanying parameters.
- Type **E** (Execute) next to the service and press Enter.
- Type **S** (Select) next to the service, specify the parameters on the Service Information panel, and press Enter.

The output for that service appears in the SYSPROG Services Output panel.

Figure 8 shows the SYSPROG Services Output panel after the DVIEW service is executed.

```

-----SYSPROG Services Output----- Row 1 of 4
COMMAND ==> DVIEW, DSN, BAB313, BMWRW. LGS1. JCL          SCROLL ==> PAGE
                                                    TARGET - SYSE
-----

15: 00: 54 CMD=DVIEW, DSN, BAB313, BMWRW. LGS1. JCL

DISK VIEW SERVICE
VOL - BAB313          BLOCK CCHHR - 0194000401
CSW - 000C1288 0C00FEF7  BLKSIZE - 264
ECB - 410234FC          SENSE - 0000

0000 C1E2D4F3 40404040 00E0C1E2 D4404040 *ASMB ..ASM *
0010 40400000 1A0F0103 00000091 079F0091 * .....j...j *
0020 098F1312 00410040 0000D3C7 E2F1F140 *.....LGS11 *
0030 40404040 C1E2D4C3 C5F2F740 0002050F * ASMCE27 .... *
0040 01020000 0091070F 0091107F 11330040 *.....j...j..... *
0050 00400000 D3C7E2F1 F2404040 4040C1E2 *...LGS12 AS*
0060 D4C5D5C7 40400000 1C0F0100 00000091 *MENG .....j *
0070 098F0091 098F2320 002E002E 0000D3C7 *...j.....LG*
0080 E2F1F140 40404040 C1E2D4C5 D5C7C1D3 *S11 ASMENGAL*
0090 0001050F 01020000 0091101F 0091101F *.....j...j... *

```

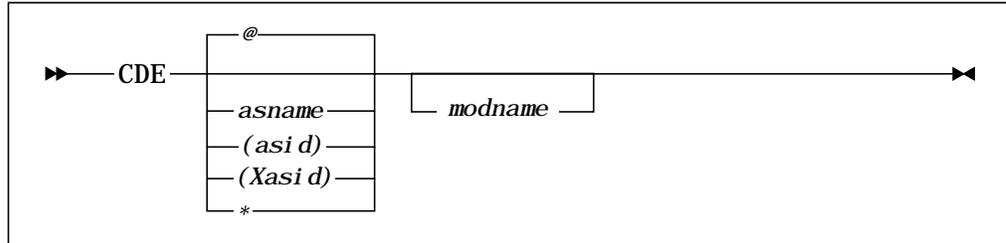
Figure 8. SYSPROG Services Output Panel for DVIEW

The SYSPROG Services Output panel displays the output for all services, with the most recently executed service positioned at the top of the display. You can scroll up or down within the panel or invoke other services by typing the appropriate name and parameters on the **COMMAND** line.

About SYSPROG Services Command Syntax

The descriptions of services, in [Chapter 4, “Synchronous Services” on page 39](#), use diagrams to illustrate command syntax. See [“Syntax Diagrams” on page viii](#) for information on how to read syntax diagrams.

For example, the syntax for the CDE service is shown in the following diagram.



Note: The examples in this section use the CDE service to demonstrate command syntax, but the syntax rules described here are applicable to all services.

The diagram indicates that the CDE service accepts two parameters: *asname*, a representation of the address space name (which can be the actual address space name, the address space identifier in decimal or hexadecimal format, * for the last address space entered, or @ for your own address space) and *modname* which, when specified, displays information about a single module.

When you invoke a service from the **COMMAND** line, in most cases you can use either a blank space or a comma to separate the service name and the parameters. (For clarity, the syntax diagrams in this book do not show commas between parameters.)

For example, this is a valid service command:

```
cde inventory
```

So is this:

```
cde, inventory
```

If you want to issue a command with the default value for a parameter, you do not have to actually type in a value for that parameter. For example, this command:

```
cde
```

displays CDE information about all modules in your own address space (since your own address space is the default for the *asname* parameter).

The CDE service also allows you to display information about a single module by specifying the *modname* parameter after the *asname* parameter, like this:

```
cde inventory ieavtsdt
```

The command above displays information about the module IEAVTSDT in the address space INVENTORY.

Using Commas as Placeholders for Default Values

If you want to display information about a specific module in your own address space (the default), you do not have to type out your address space name. However, you must insert commas as placeholders for the *asname* parameter, as shown below.

```
cde, , ieavtsdt
```

SYSPROG Services parameters are positional. This means that for some commands, SYSPROG Services expects to find *asname* as the first parameter after the service name and the *modname* as the second parameter. If you type in a value for *modname* without using commas as placeholders for *asname*, SYSPROG Services interprets the module name as an address space name and an error message may result.

Executing SYSPROG Services from an Operator Console

You can invoke SYSPROG Services from an OS/390 operator console by using the MODIFY (F) command or a predefined CMDID character.

Using the MODIFY (F) Command

Specify the MODIFY (F) command in this format:

```
F pasname, R=sysprog_service
```

where *pasname* is the name of the PAS and *sysprog_service* is the name of the service you want to invoke, along with any parameters the service may require.

Using a CMDID Character

You may define a CMDID character in the JCL for the OS/390 PAS.

If you already defined a CMDID character in the OS/390 PAS JCL, you can invoke SYSPROG Services by preceding the service name with *two* of the defined CMDID characters. The pair of characters tells the system to invoke SYSPROG Services, as R= does in the previous example. The two characters must be the same.

Type the command in this format:

```
**sysprog_service
```

where * is any valid CMDID character.

For example, if you specified CMDID=# in the OS/390 PAS JCL, you could type ##INFO at the console to invoke the INFO service.

Valid CMDID Characters: The following are valid CMDID characters. They can be specified as a single character, a single character in quotation marks, or its hexadecimal equivalent (two digits).

```
¢ . < ( + | & ! * ) ~ - / % _ > ? : # @ ' = "
```

Independent Operations

SYSPROG Services has the ability to operate independently from MainView for OS/390 as a started task (STC) or as a TSO command (in Line mode).

Executing SYSPROG Services as a Started Task

When SYSPROG Services is executed as a started task (STC), it receives commands from an operator console and sends the response to that console. Commands may be entered in response to an outstanding WTO prompt via an operator MODIFY command or via a command identification character. The specific method is determined by options specified in BBPARM member \$\$INSYS0 and the presence (or absence) of member \$\$INNWTO. See [Chapter 2, “Controlling the SYSPROG Services Environment” on page 7](#) for details.

BBSAMP member SYSPROGJ contains a sample procedure for executing SYSPROG Services as a started task that you can copy to a cataloged procedure data set. This member also contains an explanation of each DD statement contained in the procedure.

Note: It is possible to start SYSPROG Services prior to starting the Job Entry Subsystem (JES) if desired. However, if you do, all referenced cataloged data sets must be cataloged in the Master catalog. Furthermore, SYSOUT data sets cannot be used since they require JES services.

Executing Services in Started Task Mode

If you are running SYSPROG Services in Started Task mode, you can enter commands using the MODIFY (F) command. For example, to invoke the CPU service, type:

```
F j obname | ID, CPU
```

Using MODIFY (F) to enter commands in Started Task mode is recommended because

- The current outstanding reply number does not have to be found; thus, you can enter the command faster and with less chance of syntax error.
- The output from the service returns only to the console from which the service was invoked.

A console must have SYS or ALL authority before commands can be executed through MODIFY. The message CONSOLE NOT AUTHORIZED is displayed if the authority has not been assigned.

After you use MODIFY (F) to enter a command, SYSPROG Services responds with

```
AMFOODI ENTER SYSPROG Servi ces COMMAND (si d)
```

You can enter up to 16 MODIFY commands simultaneously. If you attempt to enter more than 16 MODIFY commands, the message MODI FY REJECTED TASK BUSY is displayed.

SYSPROG Services as a TSO Command

To invoke SYSPROG Services in TSO Line mode, you can perform one of the following:

- At the **TSO READY** prompt, type **SYSPROG**.
- From an ISPF **COMMAND** line, type **TSO SYSPROG**.

The following message is displayed:

```
AMTIN1I SYSPROG INITIALIZATION COMPLETE  
AMT001A SYSPROG
```

You can now issue SYSPROG Services commands. If you need to see a list of the commands, on the command line, at the **AMT001A SYSPROG** prompt, type **HELP**.

To display the last service used, at the **AMT001A SYSPROG** prompt, type **?**.

To terminate the SYSPROG Services command, at the **AMT001A SYSPROG** prompt, type **END**.

Important

In order for SYSPROG Services to run in the TSO environment, you must set the authorization through AutoCustomization, or through manual customization.

In AutoCustomization, select the step with the description *Authorize SYSPROG to run in a TSO environment*. Refer to the *MAINVIEW SYSPROG Services Customization Guide* for information on how to complete this step manually.

Part 3. SYSPROG Services Reference

This part provides detailed information about synchronous services and the Exception Monitor samplers, including proper syntax, required and optional parameters, and usage examples.

Chapter 4. Synchronous Services	37
ALLOCATE	38
APF	39
ASM	45
ASVT	48
AUTHTSO	49
BBXS	53
CDE	55
CHAP	57
CLEAR	59
COMMAND	61
CONSOLES	62
CPU	63
CSA	66
DEALLOC	68
DEVIATN	69
DONTSWAP	71
DSNAME	72
DUMP	74
DVIEW	78
EDTINFO	83
ENQUEUEES	87
EQUATE	90
ESCLASS	92
ESTORAGE	97
EXECUTE	99
EXIT	101
FINDMBR	102
INFO	104
IO	105
IPLDATA	107
LABEL	109
LCPU	110
LLIST	112
LNKLST	116
LOADLPA	122
LOG	125
LPA	128
LWAIT	130
LX	132
MCOMMAND	134
MDEVICE	135
MEMSCAN	137
MIO	142
MLCU	145
MONITOR	148
MOUNT	150

MPATH	151
MSTORAGE	153
MTP	155
NUCLEUS	156
OKSWAP	158
PAGING	160
PARMLIST	161
PIO	162
PPT	163
PRIVATE	167
PROGRESS	173
PRSM	175
REPLIES	179
RESERVES	180
RSM	182
SDUMP	186
SETOP	189
SMFEXIT	190
SMSLIST	199
SOFTFRR	201
SPACE	205
SRM	207
SSVT	210
STATUS	212
SUBMIT	215
SVCFIND	216
SYSDUMP	218
TCB	219
TERM	222
TIME	223
TIOT	224
TPIO	227
TQE	229
TRACK	232
TRTBL	235
TSULIST	238
UCB	240
USING	243
VIO	245
VMCMD	246
WARNING	247
ZAP	249
Chapter 5. Exception Monitor Samplers	251
ACPU	252
AIO	254
AIOR	256
APAG	258
APGR	260
ASRV	262
ASSR	264
BCPU	266
BIO	267

BPAG	269
BSRV	271
CHA	273
CPU	275
CQD	277
CSA	278
CSMJ	280
DEV	282
DIOQ	283
DNR	285
DOM	286
DVTN	287
ENQ	289
FIX	290
INT	291
JET	292
JSU	294
JVR	296
LOG	297
MTP	300
NRQ	301
NRQP	303
OUT	305
PAG	306
PGQ	307
PGSR	308
PSRV	310
REP	312
RES	313
RSU	314
SMF	315
SPA	317
SRM	318
SWP	319
TCPU	320
TIO	322
TPAG	324
TSRV	326
WDLY	328
WOBJ	329
WRT	330
WTO	331
WTOR	332

Chapter 4. Synchronous Services

The synchronous services offered by SYSPROG Services help improve and increase productivity by allowing you to manipulate various OS/390 internals. This chapter describes all services in detail. For information on using the Services Menu, see [Chapter 3, “Using SYSPROG Services”](#) on page 21.

Note: The services operate in an ISPF environment. Use PF3 to return to the MainView window interface.

ALLOCATE

The ALLOCATE (ALL) service allocates a particular device to prevent allocation of the device by the operating system.

Syntax

```
▶▶ ALL — dvn ◀◀
```

where

dvn Is the hexadecimal device number of the device.

Example

To allocate device 180, type:

```
ALL 180
AMTA11I  DEVICE 180 ALLOCATED
```

Note: ALLOCATE sets bits UCBALOC, UCBNLN, and UCBRESV in field UCBSTAT for the device specified.

APF

The APF (AP) service dynamically updates the APF list. Use the APF service to list, add, delete, change, or restore entries in the APF list without having to wait for the next IPL.

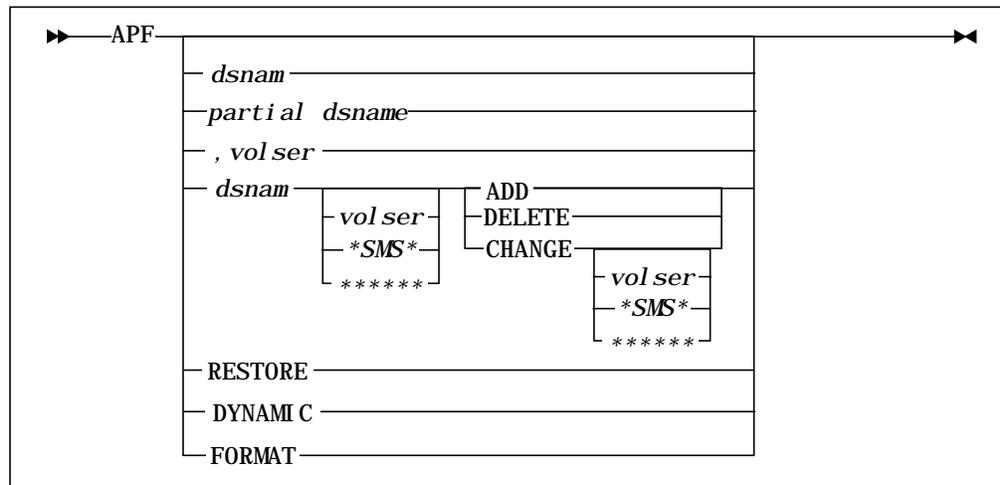
This service

- lists the current authorized program libraries and their volume serial numbers.
- selectively lists libraries by data set name, partial data set name, or volume serial number.
- adds a new entry to the APF list; authorizes link list data sets that were not authorized at IPL.
- deletes an entry from the APF list.
- changes the volume serial number for an entry in the APF list.
- changes the format of the APF list from `STATIC` to `DYNAMIC`.
- displays the APF list format (`STATIC` or `DYNAMIC`).
- restores the APF entries created by the nucleus initialization program (NIP) at IPL.

The APF service provides many features that are not available through the operating system

- the ability to change the volume serial number for an entry in the APF list.
- the ability to selectively modify the link list after adding or deleting an entry from the APF list.
- the option to omit the volser on any command. Unlike the `SETPROG` command and the `CSVAPF` macro, which require the specification of a volser, the APF service scans the catalog for the correct volser if one is not specified.
- protection for a `DYNAMIC` APF list if it was modified through the APF service.
- the ability to restore the `STATIC` APF list created by the NIP without using the `SETPROG` command.

Syntax



where

<i>dsname</i>	Is the data set name of the authorized library.
<i>partial dsname</i>	Represents the first characters of a data set name. For instance, if you type APF RXA15, the service lists every data set that has a prefix of RXA15.
<i>vol ser</i>	Is the volume serial number. All the authorized libraries that reside on this volume are displayed.
<i>dsname vol ser</i>	Is the data set name and volume serial number. <ul style="list-style-type: none"> • If a volser is not specified, the cataloged volser for the data set is used. • If *SMS* is specified as the volser, SMS will manage the data set. • If ***** is specified as the volser, the system residence volume is used.
ADD	Adds the specified data set on the specified volume to the APF list.
DELETE	Deletes the specified data set from the APF list.
CHANGE	Changes the volser on which the data set resides to the volume that follows. <ul style="list-style-type: none"> • If a volser is not specified, the cataloged volser for the data set name is used. • If *SMS* is specified as the volser, SMS will manage the data set. • If ***** is specified as the volser, the system residence volume is used.
RESTORE	Restores the NIP-created APF list, as long as that list was in STATIC format. It may not be used if the NIP created the APF list in DYNAMIC format, as DYNAMIC lists cannot be restored.
DYNAMIC	Switches the APF format from STATIC to DYNAMIC. (This is the equivalent of the IBM SETPROG APF, FORMAT=DYNAMIC command.)

FORMAT Displays the format of the APF list (STATIC or DYNAMIC) and indicates whether or not this format may be changed.

Note: The APF service creates a new APF list in SQA storage. The NIP-created APF list remains. If an error occurs, the NIP-created APF list can be restored using the RESTORE option. Changes are valid until the next IPL.

Examples

The following examples illustrate the ways to invoke the APF service.

List Example

To list all entries in the APF list, type:

```

apf
AMTA71I   APF LIST AT LOCATION 01B08000 HAS 201 ENTRIES
AMTA72I   VOLUME OPNS00     DSN RES1. RES31. BBLINK
  
```

Legend:

1. Hexadecimal address of the APF list in storage.
2. APF list.
3. Volser of the authorized library, *SMS*, or *****.
4. Data set name of the authorized library.

Note: The APF service repeats message AMTA72I for every entry in the APF list.

To list all APF entries that start with RES1, type:

```

apf res1
AMTA77I   THE FOLLOWING ENTRIES MATCHED THE REQUEST
AMTA72I   VOLUME OPNS00     DSN RES1. RES31. BBLINK
  
```

To list all APF entries for any data set on volser OPNS00, type:

```

apf, opns00
AMTA72I   VOLUME OPNS00     DSN RES1. RES30. BBLINK
  
```

Note the space between APF and the comma.

Add Example

To create a new entry in the APF list for RES1.RES30.BBLOAD, residing on OPNS01, type:

```
apf res1.res30.bbload opns01 add
AMTA73I  APF LIST NOW CONTAINS ENTRY FOR
AMTA72I  VOLUME OPNS01    DSN RES1.RES30.BBLOAD
AMTAXGA  ENTER Y TO SET UP AUTHORIZATION IN LINKLIST FOR DATA SET
```

To create a new entry in the APF list for a cataloged data set, type:

```
apf res1.res30.bbload, , add
AMTA73I  APF LIST NOW CONTAINS ENTRY FOR
AMTA72I  VOLUME OPNS01    DSN RES1.RES30.BBLOAD
AMTAXGA  ENTER Y TO SET UP AUTHORIZATION IN LINKLIST FOR DATA SET
```

The APF service determines where RES1.RES30.BBLOAD is cataloged and creates a new APF list with the entered data set and the cataloged volser.

To add a data set to the APF list and have the data set managed by SMS, type:

```
apf res1.user.file *sms* add
AMTA73I  APF LIST NOW CONTAINS ENTRY FOR
AMTA72I  VOLUME *SMS*    DSN RES1.USER.FILE
AMTA7MI  SMS MANAGED DATASET RES1.USER.FILE
AMTAXGA  ENTER Y TO SET UP AUTHORIZATION IN LINKLIST FOR DATA SET
```

Note: If you do not want the data set to appear in the link list, type any character other than Y.

Delete Example

To remove an entry from the APF list, type:

```
apf res1.res30.bbload opns01 delete
AMTA74I  APF LIST ENTRY DELETED FOR
AMTA72I  VOLUME OPNS01    DSN RES1.RES30.BBLOAD
```

Change Example

To change the volser of an existing APF entry, type:

```
apf res1.res30.bbload opns01 change perf01
AMTA75I  APF LIST ENTRY CHANGED. OLD VOLUME SER OPNS01 IS NOW
AMTA72I  VOLUME PERFO1    DSN RES1.RES31.BBLOAD
AMTA7JI  ENTER Y TO CONTINUE WITH AUTHORIZATION REQUEST
```

Restore Example

To restore the APF list to the original list created by NIP at IPL, type:

```
apf restore  
AMTA79I THE NIP-CREATED APF LIST HAS BEEN RESTORED
```

Only STATIC NIP-created APF lists may be restored. If you try to use the APF RESTORE command on a NIP-created DYNAMIC list, this message is displayed:

```
apf restore  
AMTAXSP NIP-CREATED DYNAMIC APF LIST CANNOT BE RESTORED; CONTACT IBM
```

If you try to restore a STATIC APF list that has not changed since it was created by the NIP, this message is displayed:

```
apf restore  
AMTAXNL CURRENT APF LIST HAS NOT BEEN CHANGED; NO NEED TO RESTORE
```

Dynamic Example

To change the format of the APF list from STATIC to DYNAMIC, type:

```
apf dynamic  
AMTAXCF THE APF LIST FORMAT HAS BEEN CHANGED FROM STATIC TO DYNAMIC
```

To use the APF DYNAMIC command, DFSMS/MVS must be installed on your system. If it is not, this message is displayed:

```
apf dynamic  
AMTAXEV APF FORMAT CANNOT BE CHANGED. DFSMS/MVS IS NOT INSTALLED.
```

Format Example

To display the format of the current APF list, type:

```

apf format
AMTAXFI   APF LIST IS IN ❶STATIC FORMAT.
  
```

Legend:

1. Format of the APF list: STATIC or DYNAMIC.

If the NIP created the APF list in DYNAMIC format, the following two messages are displayed:

```

apf format
AMTAXFI   APF LIST IS IN DYNAMIC FORMAT.
AMTAXNA   STATIC APF FORMAT NOT AVAILABLE.
  
```

A STATIC list is never available if the NIP-created list was DYNAMIC. However, suppose the NIP-created list was STATIC but the APF list is currently DYNAMIC. You would see these messages:

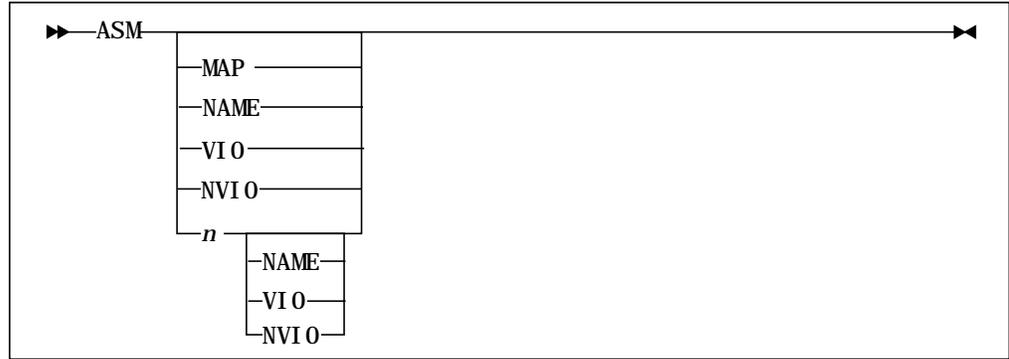
```

apf format
AMTAXFI   THE APF LIST IS IN DYNAMIC FORMAT.
AMTAXAS   STATIC APF FORMAT AVAILABLE.
  
```

ASM

The ASM service provides information for managing auxiliary storage and paging data sets.

Syntax



where

- MAP** Provides a detailed display of the VIO and non-VIO page slots owned by each job.
- n** Produces MAP output for address spaces that own n or more non-VIO page slots.
- NAME** Sorts MAP output by address space name in alphabetical order.
- VIO** Sorts MAP output in descending order by the number of VIO slots per address space. The 10 highest users of VIO slots are displayed.
- NVIO** Sorts MAP output in descending order by the number of non-VIO slots per address space. The 10 highest users of non-VIO slots are displayed.

Examples

To display information about auxiliary storage and paging data sets, type:

```
asm
AMTA5CI LAST IPL WAS COLD START (CLPA)
AMTA51I ASMVT 00FD9C30 ①
AMTA52I TOTAL LOCAL SLOTS 490650 AVAILABLE 353914 = 72 percent ②
AMTA5DI LARGEST HOLDER OF SLOTS IS PAS WITH 1% ③
AMTA53I D/S 0 ( 251 BOOLE1) PLPA SIZE 2600 FREE 1006 = 38% BURST 10 ④
AMTA53I D/S 1 ( 251 BOOLE1) COMN SIZE 1300 FREE 873 = 67% BURST 10
AMTA56I D/S 2 (NOT IN USE) DUPLX ⑤
AMTA53I D/S 3 ( 541 BABPO1) *LOCAL SIZE 13650 FREE 12308 = 90% BURST 30
AMTA53I D/S 4 ( 544 BABPO2) LOCAL SIZE 13650 FREE 12113 = 89% BURST 30
AMTA56I D/S 5 (NOT IN USE) LOCAL
AMTA57I D/S 6 (NOT IN USE BECAUSE OF ERRORS) ⑥
AMTA58I * = NOT ELIGIBLE FOR VIO USE ⑦
```

Legend:

1. Hexadecimal address of the ASMVT.
2. Total number of page slots for user paging, the actual number of slots currently free, and the percentage of slots currently free.
3. Address space holding the largest number of slots.
4. A line is displayed for each page data set.

For each active data set, MV MANAGER for MVS message AMTA53I displays the

- Device and volume serial numbers
 - Usage class (such as PLPA, LOCAL, DUPLX, COMN, or UNKN)
 - Size of the data set in page slots
 - Percentage of page slots currently available
 - Actual number of page slots currently available
5. Message AMTA56I is issued for a page data set when SYS1.PARMLIB member IEASYSxx
 - Defines the page data set but the data set was not available during an IPL
 - Does not include the DUPLEX parameter, or the DUPLEX parameter specifies a data set that was not available during an IPL (the system is always prepared for a duplex data set)
 6. Message AMTA57I identifies page data sets that were deactivated because of I/O errors.
 7. An asterisk denotes page data sets that are not eligible for VIO paging.

To list the number of VIO and non-VIO page slots owned by each job, type:

```
asm map

AMTA5CI LAST IPL WAS COLD START (CLPA)
AMTA51I ASMVT 00FD9C30
AMTA52I TOTAL LOCAL SLOTS 490650 AVAILABLE 353914 = 72%
AMTA5DI LARGEST HOLDER OF SLOTS IS PAS WITH 1%
AMTA53I D/S 0 ( 251 BOOLE1) PLPA SIZE 2600 FREE 1006 = 38% BURST 10
AMTA53I D/S 1 ( 251 BOOLE1) COMN SIZE 1300 FREE 873 = 67% BURST 10
AMTA56I D/S 2 (NOT IN USE) DUPLX
AMTA53I D/S 3 ( 541 BABP01)*LOCAL SIZE 13650 FREE 12308 = 90% BURST 30
AMTA53I D/S 4 ( 544 BABP02) LOCAL SIZE 13650 FREE 12113 = 89% BURST 30
AMTA56I D/S 5 (NOT IN USE) LOCAL
AMTA57I D/S 6 (NOT IN USE BECAUSE OF ERRORS)
AMTA58I * = NOT ELIGIBLE FOR VIO USE

      ①                ②                ③
AMTA54I GOJOB OWNS 0 VIO 295 NON-VIO SLOTS
AMTA54I INVENTORY OWNS 24 VIO 106 NON-VIO SLOTS
```

Legend:

1. A line is displayed for each address space in the system that owns slots.
2. Number of external page slots owned by this address space for VIO activity.
3. Number of external page slots owned by this address space for non-VIO activity.

To list the 10 highest users of non-VIO slots who own at least 1000 non-VIO slots, type:

```
asm 1000 nvi o

AMTA5CI LAST IPL WAS COLD START (CLPA)
AMTA51I ASMT 00FDA230
AMTA52I TOTAL LOCAL SLOTS 486000 AVAILABLE 342356 = 70%
AMTA5DI LARGEST HOLDER OF SLOTS IS EYUX110 WITH 1%
AMTA53I D/S 0 ( B14 PAGEB2) PLPA SIZE 15000 FREE 9069 = 60% BURST 30
AMTA53I D/S 1 ( B14 PAGEB2) COMN SIZE 15000 FREE 13377 = 89% BURST 30
AMTA56I D/S 2 (NOT IN USE) DUPLX
AMTA53I D/S 3 ( B17 PAGEB3) LOCAL SIZE 132000 FREE 95361 = 72% BURST 30
AMTA53I D/S 4 ( B14 PAGEB2) *LOCAL SIZE 90000 FREE 52890 = 59% BURST 30
AMTA53I D/S 5 ( B21 PAGEB4) *LOCAL SIZE 132000 FREE 98310 = 74% BURST 30
AMTA53I D/S 6 ( B20 PAGEB6) *LOCAL SIZE 132000 FREE 95770 = 73% BURST 30
AMTA58I * = NOT ELIGIBLE FOR VIO USE

AMTA54I EYUX110 OWNS 0 VIO 5489 NON-VIO SLOTS
AMTA54I CVMFBC1 OWNS 0 VIO 4066 NON-VIO SLOTS
AMTA54I VLF OWNS 0 VIO 3348 NON-VIO SLOTS
AMTA54I PAS OWNS 0 VIO 2961 NON-VIO SLOTS
AMTA54I PMOPROC OWNS 0 VIO 2565 NON-VIO SLOTS
AMTA54I CVMFBT1 OWNS 0 VIO 2126 NON-VIO SLOTS
AMTA54I CNMPRB22 OWNS 0 VIO 1912 NON-VIO SLOTS
AMTA54I DB2EDBM1 OWNS 0 VIO 1846 NON-VIO SLOTS
AMTA54I DEC3 OWNS 0 VIO 1798 NON-VIO SLOTS
AMTA54I DJMC4 OWNS 0 VIO 1643 NON-VIO SLOTS
```

Legend:

1. Output is sorted in descending order by the number of non-VIO slots.

ASVT

The ASVT service displays information for three address space queues maintained in the ASVT control block. They are:

- Available queue
- Non-reusable replacement queue
- Start/SASI queue

Each queue displays the following information:

- Current number of slots (ASIDs) on the queue
- Percentage of original number of slots (ASIDs) on the queue
- Original number of slots (ASIDs) on the queue
- Name of system parameter that defined the original queue length

When the system marks an ASID as non-reusable, it replaces the ASID by transferring a slot from the non-reusable replacement queue to the available queue until the replacement queue is empty. If an attempt is made to start a started task and the available queue is empty, an ASID on the Start/SASI queue is used.

Syntax



Example

To display information for the three address space queues, type:

asvt					
AMTAV1I	Description	Current	Percent	Original	Parameter
AMTAV2I	-----	-----	-----	-----	-----
AMTAV3I	Available queue	389	77.80	500	MAXUSER
AMTAV3I	Non-reusable replacement queue	19	76.00	25	RSVNONR
AMTAV3I	Start/SASI queue	25	100.00	25	RSVSTRT

AUTHTSO

The AUTHTSO (AU) service

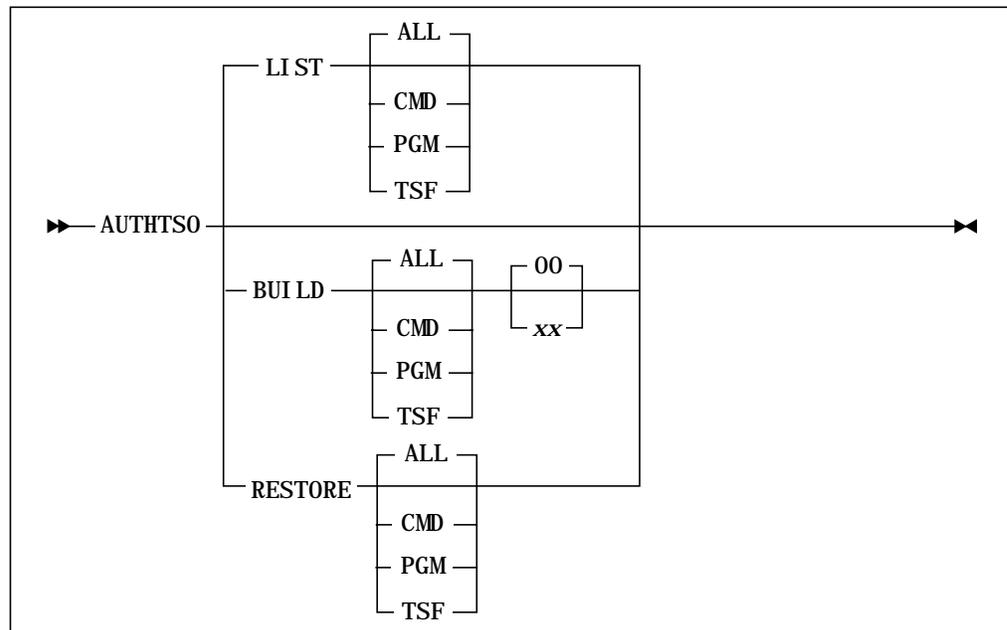
- Lists the contents of the current authorization lists
- Builds a new authorization list from any IKJTSOxx member in SYS1.PARMLIB
- Restores the authorization list built at IPL

Use the AUTHTSO service to modify the contents of the TSO authorization lists for the duration of the current IPL. This service eliminates the need to IPL and lets you test updates before entering them into production. The AUTHTSO service is valid for TSO/E Version 1.4 or later.

TSO authorization lists define which TSO commands or programs can be invoked with APF authorization. AUTHTSO accesses each TSO authorization list independently of the other lists. The three types of TSO authorization lists are as follows:

AUTHCMD (CMD)	Authorized TSO commands
AUTHPGM (PGM)	Authorized TSO programs
AUTHTSF (TSF)	APF-authorized programs that can be called through the TSO Service Facility (TSF)

Syntax



where

LIST Lists the contents of the current authorization list.

BUILD	Builds a new authorization list from any IKJTSoxx member in SYS1.PARMLIB; 00 is the default.
RESTORE	Restores the authorization list built at the time of IPL.
ALL	Applies the appropriate command to all three lists: TSO commands, programs, and Service Facility.
CMD	Specifies authorized TSO commands. You can substitute AUTHCMD, IKJEFTE2, or E2 for CMD.
PGM	Specifies authorized TSO programs. You can substitute AUTHPGM, IKJEFTE8, or E8 for PGM.
TSF	Specifies the TSO Service Facility. You can substitute AUTHTSF, IKJEFTAP, or AP for TSF.
xx	Is the suffix of SYS1.PARMLIB member IKJTSoxx.

Examples

The following examples illustrate the ways to invoke the AUTHTSO service.

List Examples

To display the authorization list of TSO commands (CMD), type:

```

authtso list cmd
      ①
AMTAT7I AUTHORIZED COMMAND LIST (IKJEFTE2) -
      ②      ③
AMTAT7I BUILT DURING IPL AT 02:16 ON 96.015:
AMTATZI
AMTAT8I RECEIVE TRANSMIT XMIT LIST ④
AMTAT8I SEND RACONVRT SYNC RESOLVE④

```

Legend:

1. Type of authorization list (COMMAND, PROGRAM, or APF- TSF).
2. Process by which the authorization list was built (IPL or AUTHTSO).
3. Time and date the authorization list was created.
4. Commands in the specific authorization list.

To display all authorization lists, type:

```

authtso

AMTAT7I AUTHORIZED COMMAND LIST (IKJEFTE2) -
AMTAT7I BUILT DURING IPL AT 02:16 ON 96.022:
AMTATZI
AMTAT8I RECEIVE TRANSMIT XMIT LISTD
AMTAT8I SEND RACONVRT SYNC RESOLVE
AMTATZI
AMTAT7I AUTHORIZED PROGRAM LIST (IKJEFTE8) -
AMTAT7I BUILT DURING IPL AT 02:16 ON 96.022:
AMTATZI
AMTAT8I IEBCOPY SPFCOPY IKJEFF76 RESOLVE
AMTAT8I ICHUT100 ICHUT200 ICHUT400
AMTATZI
AMTAT7I AUTHORIZED APF-TSF LIST (IKJEFTAP) -
AMTAT7I BUILT DURING IPL AT 02:16 ON 96.022:
AMTATZI
AMTAT8I IEBCOPY SPFCOPY IKJEFF76

```

Build Example

Refer to the appropriate IBM manual for syntax rules for the IKJTSoxx member.

To build a new authorization list of TSO commands from SYS1.PARMLIB member IKJTSo01, type:

```

authtso build ikjefte2 01

AMTAT9A REPLY Y TO CONFIRM "BUILD" OF TSO CMD LIST, N TO CANCEL
y
AMTATBI TSO CMD LIST (IKJEFTE2) BUILT

```

Restore Example

To restore all authorization lists to their status at IPL, type:

```

authtso restore all

AMTAT9A REPLY Y TO CONFIRM RESTORE OF TSO CMD LIST, N TO CANCEL
y
AMTATBI TSO CMD LIST (IKJEFTE2) RESTORED
AMTAT9A REPLY Y TO CONFIRM RESTORE OF TSO PGM LIST, N TO CANCEL
y
AMTATBI TSO PGM LIST (IKJEFTE8) RESTORED
AMTAT9A REPLY Y TO CONFIRM RESTORE OF TSO TSF LIST, N TO CANCEL
y
AMTATBI TSO TSF LIST (IKJEFTAP) RESTORED

```

Usage Notes

- The AUTHTSO service prompts you for a response before it updates each of the three authorization lists. To restore the original authorization list, type Y in response to message AMTAT9A.
- To use the original authorization list, you must log off and then log back on to TSO when AUTHTSO completes the restore. TSO creates a static copy of the authorization list in your TSO address space when you log on.
- Prior to TSO/E Version 1.3, authorization lists were implemented as lists in load module IKJTABLS. Each table was located by way of an entry point. The list names in IKJTSOxx and their corresponding entry points in IKJTABLS are as follows:

List Name	Entry Point
AUTHCMD	IKJEFTE2
AUTHPGM	IKJEFTE8
AUTHTSF	IKJEFTAP

The entry point names are retained in the list headers built from IKJTSOxx.

BBXS

The BBXS (BBX) service displays the status of the BMC Software subsystem services.

When diagnosing problems, BMC Software Support personnel may ask you to use the BBXS service to verify information about the subsystem.

Syntax



where

- BBXS** Displays information about the BBXS subsystem; the default.
- subsystemID* Displays information about the specified BMC Software subsystem service (when it is named something other than BBXS).

Examples

To display information about the BBXS subsystem, type:

```

bbxs
AMTBXAD BBX Subsystem Information
AMTBXDD
=====
AMTBX2D Subsystem name=BBXS; SSCT=00BD3270 ## AUTOINIT ##
AMTBX3D BBCT addr=00B5A000; restart count=01; RMID=BPB1160 , LEVEL=0005
AMTBX5D Initialized by ITSTNPAS (mod=CX10SDVR, key=4) at 03:49:32 on 12/17/95
AMTBX6D BBX was loaded from BB.ITSTN.BBLINK
AMTBX4D Previous BBCT Addr=00B9F000; RMID=BPB1144 , level=0005 ## AUTOINIT ##
AMTBX5D Initialized by DCSBBI (mod=ASTXA1MN, key=8) at 03:47:48 on 12/17/95
AMTBX6D BBX was loaded from SYS0.SBBPROD.BBLINK
    
```

Legend:

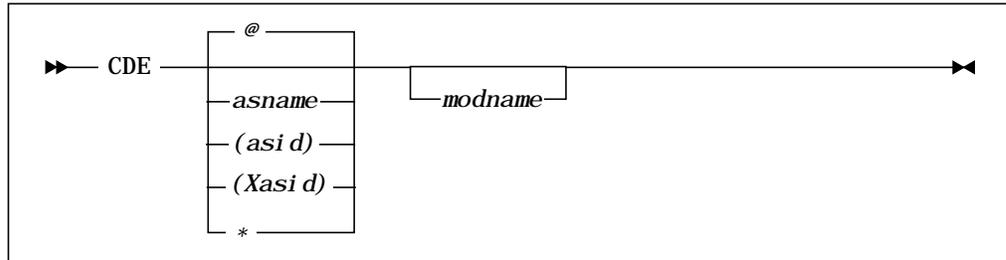
1. Name of the BMC Software subsystem service currently running.
2. Address of the Subsystem Services Control Table (SSCT).
3. Address of the Boole & Babbage Subsystem Services Control Table (BBCT).
4. Number of times the subsystem was restarted.
5. RMID of the subsystem.

6. Name of the address space that created the subsystem.
7. Module or program that created the subsystem.
8. TCB key of the address space that created the subsystem.
9. Time the subsystem was created.
10. Date the subsystem was created.
11. Data set from which the subsystem modules were loaded.
12. Information about the previously loaded BBCT.

CDE

The CDE service lists all the modules loaded in a specified address space and displays information about them from the directory entries.

Syntax



where

- @ Specifies your own address space; the default.
- asname Is the address space name.
- (asi d) Is the address space identifier in decimal format.
- (Xasi d) Is the address space identifier in hexadecimal format.
- * Specifies the last address space entered.
- modname Limits the display to the specified module.

Examples

To display information about modules in address space INVENTORY, type:

```

cde inventory

AMTY40I
AMTY42I TCB ADDRESS IS 7E2278
AMTY43I LISTING OF ASSOCIATED CDE' S
AMTY51I ① ② ③ ④ ⑤ ⑥ ⑦ ⑧ ⑨ ⑩ ⑪ ⑫
AMTY45I CDE-ADDR NAME ENTRY-PT USE SP RN RU APF AC1 JPA LPA MI
AMTY46I 007DC678 ISTINCTS 838CBD80 0001 251 N Y Y N Y N N
  
```

Legend:

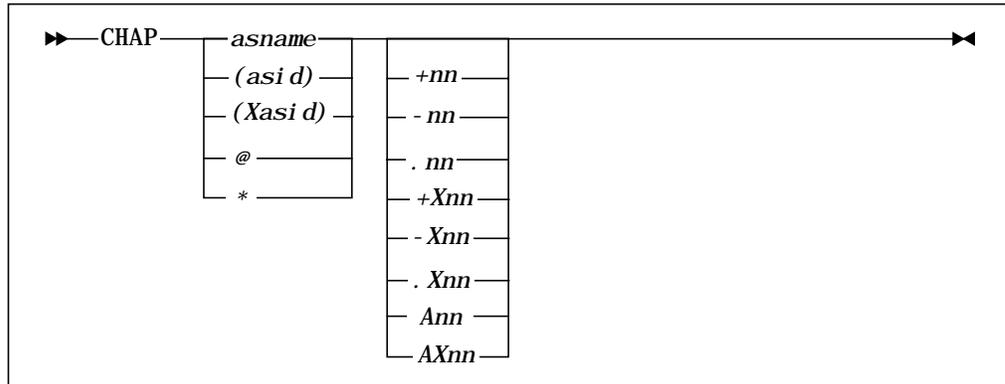
1. Address of the Contents Directory Entry (CDE).
2. CDE name.
3. Module entry point address.
4. Module use count.
5. Subpool ID.

6. Whether the module is re-enterable.
7. Whether the module is serially reusable.
8. Module has been loaded from an authorized library.
9. Whether a program authorization flag exists.
10. Whether the module is in the job pack area.
11. Whether the module is loaded in the link pack area (LPA). (Y if it is loaded in LPA; N if it is part of the nucleus.)
12. Whether a minor CDE flag exists.

CHAP

The CHAP (CHA) service helps you manage jobs by changing the internal dispatching priority of an address space.

Syntax



where

- asname* Is the address space name.
- (asi d)* Is the address space identifier in decimal format.
- (Xasi d)* Is the address space identifier in hexadecimal format.
- @ Specifies your own address space.
- * Specifies the last address space entered.
- +nn* Adds *nn* to the current dispatching priority of the address space.
- nn* Subtracts *nn* from the current dispatching priority of the address space.
- . nn or Ann* Sets *nn* as the absolute priority of the address space.
Note: If your terminal does not allow you to use the period (.), use the character A in place of the period.
- +Xnn* Adds hex *nn* to the current dispatching priority of the address space.
- Xnn* Subtracts hex *nn* from the current dispatching priority of the address space.
- . Xnn or AXnn* Sets hex *nn* as the absolute priority of the address space.
Note: If your terminal does not allow you to use the period (.) to set this value, use the character A in place of the period.

Examples

To increase the internal dispatching priority of address space INVENTORY by 3, type:

```

chap inventory +3
AMTX11I  JOB 201 INVENTORY STEP1  PRTY AF (175)  PGP 2/ 1
AMTX11I  JOB 201 INVENTORY STEP1  PRTY DF (223)  PGP 2/ 1

```

Legend:

1. JES job ID.
2. Jobname.
3. Current stepname.
4. Old dispatching priority.
5. Performance group and period.
6. New dispatching priority.

If you do not specify a new dispatching priority, you are prompted for one. For example, to change the absolute priority of address space INVENTORY to 253, type:

```

chap inventory
AMTX11I  JOB 201 INVENTORY STEP1  PRTY AF (175)  PGP 2/ 1
AMTX12A  ENTER CHANGE VALUE AS +NN, -NN, .NNN, OR ANNN FOR ABSOLUTE
. 253
AMTX11I  JOB 201 INVENTORY STEP1  PRTY FD (253)  PGP 2/ 1

```

Legend:

1. Priority change, as typed by the user.

Usage Notes

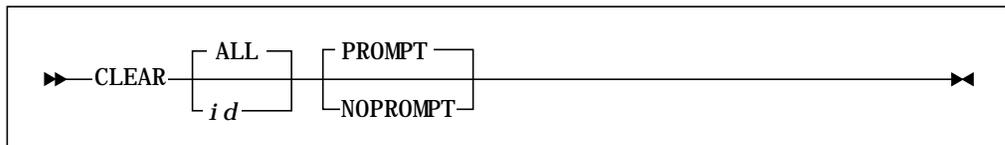
- The CHAP service changes the priority of the entire address space (ASCBCHAP), which effectively changes all tasks in the address space.
- The priority change remains in effect only for the life of the current job step.
- If you type END when you are prompted for the priority change, the service ends with no priority change.

CLEAR

The CLEAR (CLE or CLR) service clears system dump data sets without printing the information. Use the CLEAR service in conjunction with the SDUMP service (which takes an SVC dump of one or more address spaces) and the SYSDUMP service (which lists the SVC dump data sets). See “SDUMP” on page 188 and “SYSDUMP” on page 220 for more information.

Note: System-generated dump data set names are not supported by this service.

Syntax



where

- ALL** Clears all system dump data sets; the default.
- id** Is a one- or two-digit number representing the system dump data set to be cleared.
- PROMPT** Lists the title of the dump contained in each full dump data set processed. In addition, a confirmation message is issued.

Valid responses to the confirmation message are:

- Y** Clears the system dump data set.
- E** Terminates the CLEAR service.

Typing any character other than Y or E causes the current system dump data set to be bypassed and the next full system dump data set (if any) to be processed.

Example

To clear the system dump data set #01, type:

```

clear 01
AMTC50I  SYS1.DUMP01 WAS FILLED AT 10:20 ON JAN 21, 1996 ①
AMTC51I  SOURCE=SVCDUMP ②
AMTC52I  TITLE=LOW CORE OVERLAY (LOOP) ③
AMTC50A  ENTER Y TO CLEAR SYS1.DUMP01 ④
y
AMTC53I  'SYS1.DUMP00' DATA SET CLEARED ⑤
AMTC5AI  CLEAR      PROCESSING COMPLETE
    
```

Legend:

1. When the specified dump was taken.
2. System component that took the dump.
3. Title of the dump.
4. Operator verification is required before the specified data set is cleared.
5. Dump data set has been successfully cleared.

Usage Notes

- You can omit the first operand without using delimiting commas.
- You can abbreviate the operands: A for ALL, N for NOPROMPT, and Y for PROMPT.

COMMAND

The COMMAND (COM) service executes operator commands from a TSO terminal. This service helps you solve system problems from a TSO terminal, rather than an operator’s console.

Syntax

```
▶▶ COMMAND — command —————▶▶
```

where

command Is the operator command.

Example

To execute the command V 052,ONLINE, type:

```
command v 052, online
AMTC21I      COMMAND V 052, ONLINE EXECUTED FROM CN (01)
```

Legend:

1. Determines authority necessary for the command to be executed.

All JES commands must be entered from the master console. This field displays the location from which the command was executed. Possible values are:

CN (*xx*) Console; *xx* is the console ID.

TS (*xx*) TSO terminal; *xx* is the TSO address space ID.

Usage Note

Output from most commands appears at a TSO terminal. Some commands are not designed for foreground operation and the output appears only at the master console. When the COMMAND service is used from a system console, the command execution output is displayed on the master console.

CONSOLES

The CONSOLES (CON) service displays the number of undisplayed messages per system console and the number of reply buffers in use. This service lets you determine which console in an MCS environment is not operational.

With this service, you can also determine the number of console reply buffers in use when diagnosing the cause of a wait state (for example, when a job that must issue a WTOR is waiting for reply buffers to be freed by other tasks in the system).

Syntax

```
▶▶ CONSOLES ◀◀
```

Example

To list the number of messages to be displayed, type:

```

consoles
      1      2
AMTC32I  CONSOLE 0A0 HAS 1 MESSAGES TO BE DISPLAYED
AMTC32I  CONSOLE 0A2 HAS 198 MESSAGES TO BE DISPLAYED
      3
AMTC33I  198 OF 200 CONSOLE BUFFERS IN USE
      4
AMTC34I  1 OF 006 REPLY BUFFERS IN USE
  
```

Legend:

1. Console address.
2. Number of messages to be displayed; a high number indicates a backlog of messages.
3. Total number of console buffers in use by the system; this number can be less than the sum of messages to be displayed at the various consoles because a message can be routed to more than one console.
4. Number of reply buffers in use.

CPU

The CPU service provides information about job activity by noting the proportionate use of CPU resources by job. Use CPU to monitor system activity.

Syntax



where

time Is the length of the sample period in seconds; the default is 10 seconds.

Example 1 (for PR/SM Systems)

To display information about CPU usage for the past 10 seconds, type:

```

cpu
AMTOB1I PLEX BUSY          59.1%  ❶
AMTOB1I PLEX WAIT         37.2%  ❷
AMTOB1I PLEX OVERHEAD     3.7%   ❸
                               ❹
AMTOB1I PARTITION RCVD.   34.1% WHICH IS
                               ❺
AMTOB1I                    56.8% OF ITS RELATIVE SHARE
AMTOB0I
AMTOB3I
                               ❻   ❼   ❽   ❾
AMTOB4I  JOB GEN1CLS      16.4%  A3 (163)
AMTOB4I  STC AAOSSEB2     9.2%   A4 (164)
AMTOB4I  TSO CMR4X        5.2%   FF (255)
AMTOB4I  JOB PWW1JOB      2.9%   A2 (162)
AMTOB4I  STC JES2         2.3%   EF (239)
AMTOB4I  STC DCSPAS       2.0%   FF (255)
AMTOB4I  TSO LGS171       1.7%   FF (255)
AMTOB4I  STC *MASTER*     1.7%   FF (255)
AMTOB4I  TSO EUK2         1.7%   FF (255)
AMTOB4I  STC ITSTPAS      1.5%   A0 (160)
AMTOB4I  STC CATALOG      1.4%   FF (255)
AMTOB4I  TSO IXR2         1.1%   FF (255)
AMTOB4I  STC MIMB         1.1%   FF (255)
                               ❿
AMTOB5I ALL OTHERS USED   8.6%
                               ⓫
AMTOB6I TOTAL USED       56.8%
AMTOB0I
AMTOB2I BATCH= 34.2%, STC= 43.9%, TSO= 21.9%, TOTAL=100.0% ⓬
    
```

Legend:

1. Percentage of the sample period that the complex was performing useful work.

2. Percentage of the sample period the processors were not dispatched to a partition because the partitions were waiting.
3. Percentage of the sample period used by the hardware to dispatch processors to partitions.
4. Percentage of the sample period that the partition hosting the SYSPROG Services session was dispatched. Subtracting this percentage from the PLEX BUSY percentage yields the percentage of the sample all other partitions in the complex were dispatched.
5. Percentage of the partition's *relative share* (the amount of processing time the partition is supposed to receive when the partition's demands are high and resources are limited) used during the sample period. When the partition can use more time and time is available, it can exceed 100 percent of its relative share. Likewise, the partition can (when wait assist is off) receive less than its relative share.
6. Type of address space: a batch job (JOB), started task (STC), or TSO session (TSO).
7. Names of the 15 highest usage address spaces in the system with CPU usage of at least 1%.
8. CPU usage by address space.
9. Priority of the address space.
10. CPU usage for all other address spaces in the system.
11. Sum of CPU usage for all address spaces in the partition during the sample interval. This sum is equal to the partition's relative share usage (56.8 percent in this example).
12. Usage distribution among batch jobs, started tasks, and TSO sessions.

Example 2 (for Non-PR/SM, Non-MDF Systems)

To display information about CPU usage for the past 10 seconds, type:

cpu					
AMT015I	STATISTICS BEING GATHERED FOR CPU DATA				①
AMT014I	JOB INVENTORY USED	11.96 SECS,	48%,	PRTY 11	(162)
AMT014I	JOB SMU1RLA USED	6.44 SECS,	26%,	PRTY 7	(84) ②
AMT011I	TOTAL FOR MVS 0/HEAD	4.12 SECS,	21%	③	
AMT012I	TOTAL FOR BATCH JOBS	10.01 SECS,	50%	④	
AMT013I	TOTAL FOR TSO SESSIONS	5.87 SECS,	29%	⑤	
AMT017I	CPU 0 WAS 66% BUSY				⑥

Legend:

1. Message issued at the beginning of data collection.
2. Amount of time and percentage of CPU used by each of the 10 most active jobs in the system. Although only the 10 most active jobs are displayed, the total figures reflect all jobs.
3. Total OS/390 overhead not allocated to a specific address space.
4. Total time and percentage of CPU taken by batch jobs during the sample period. This includes both submitted jobs (JOB) and started tasks (STC).
5. Total time and percentage of CPU time taken by TSO sessions during the sample period.

6. Percentage of the CPU that was busy during the sample period. (This message is repeated for each CPU in a multiprocessor.)

Note: In an MP environment, the percentages could total more than 100 percent. Over the sample period, any CPU in an MP system could have been active 100 percent of the time, or a single job could have executed on each of the CPUs several times during the sample period.

CSA

The CSA service displays common service area (CSA) and extended common service area (ECSA) utilization.

Syntax



where

MAP Provides a detailed display of CSA/ECSA usage by storage subpool and protection key.

Example

To display a detailed description of CSA and ECSA usage, type:

```

csa map
AMTSQCI STORAGE TOTAL ..... CSA : 3864K          ECSA : 15M          ❶
AMTSQCI LARGEST UNALLOCATED AREA ...: 1388K          3120K          ❷
AMTSQCI (ALLOCATED): 2092K ( 54%)          12M ( 76%) ❸
AMTSQCI (FFS=FRAGMENTED FREE SPACE): 162K ( 4%)          415K ( 3%) ❹
AMTSQCI (CONVERTED TO SQA): 384K ( 10%)          0K ( 0%) ❺
AMTSQCI CURRENTLY USED .....: 2476K ( 64%)          12M ( 79%) ❻

AMTSQ8I OWNER SUP SCHED VSPC          DM VTAM IMS USER          ❷
AMTSQ4I SP/KEY 0 1 2 3 4 5 6 7 8-F TOTAL ❸
AMTSQAI ..... CSA .....
❹          ❺          ❻
AMTSQ5I 227 9K          30K          39K
AMTSQ5I 228 14K          8K 1K          50K 170K
AMTSQ5I 231 30K 31K          67K 569K 697K
AMTSQ5I 241 736K 52K          16K 5K 7K 68K 141K 1024K
AMTSQ5I FFS 60K 20K          7K 3K 24K 23K 25K 162K ❿
AMTSQ5I TOT 848K 112K          24K 8K 128K 756K 216K 2092K ⓫
AMTSQAI ..... EXTENDED CSA .....
AMTSQ5I 227 5K          189K          231K          425K
AMTSQ5I 228 1204K          1800K          57K 18K          1132K 4211K
AMTSQ5I 231 409K 32K          130K          847K 1K          1419K
AMTSQ5I 241 4928K 112K          236K          6K 372K 5K 393K 6051K
AMTSQ5I FFS 158K 20K          106K          5K 96K 6K 23K 415K ❿
AMTSQ5I TOT 6704K 164K          2460K          68K 1546K 12K 1548K 12M ⓫
    
```

Legend:

1. Total amount of CSA/ECSA defined.
2. Largest unallocated area in CSA/ECSA.
3. Amount of space allocated and percentage of total CSA/ECSA.

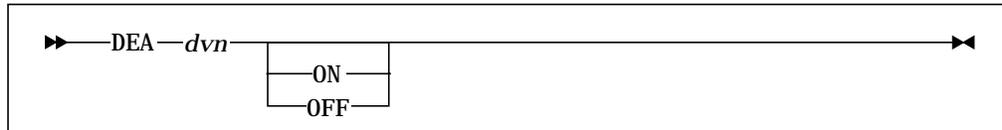
4. Amount of fragmented free space and percentage of total CSA/ECSA.
5. Amount of space converted to SQA/ESQA and percentage of total CSA/ECSA.
6. Total amount of space currently used and percentage of total CSA/ECSA.
7. Owner of storage protect key.
8. Storage protect key.
9. Storage subpool number.
10. Amount of CSA/ECSA allocated for each storage protect key within each subpool; for example, 30KB of subpool 227 is assigned a protection key of 6.
11. Total amount of CSA/ECSA currently allocated to a subpool.
12. Total amount of fragmented free space by storage protect key and subpool.
13. Total amount of CSA/ECSA currently allocated to a storage protect key and the subpool.

DEALLOC

The DEALLOC (DEA) service

- Deallocates devices allocated by service ALLOCATE
- Deallocates devices erroneously allocated or those that remain allocated by a job that does not terminate properly

Syntax



where

<i>dvn</i>	Is the hexadecimal device number of the device.
ON	Causes the deallocated device to be marked online upon completion of the service DEALLOC.
OFF	Causes the deallocated device to be marked offline upon completion of the service DEALLOC.

Example

To deallocate device 180, type:

```
DEA 180
AMTA21I  DEVI CE 180 DEALLOCATED
```

Usage Note

- The deallocated devices are online upon completion of service DEALLOC unless OFF is specified, in which case the device is deallocated and marked offline. Specifying OFF does not cause the automatic unloading of a device.
- In a shared tape environment, DEALLOC followed by VARY OFFLINE unloads a device that may be in use on another system.
- The DEALLOC service only resets bits in the UCB. No measures are taken to avoid adverse effects on active tasks using the UCB.

DEVIATN

The DEVIATN (DEV) service displays deviation from a list of jobs that must be active during specified time intervals and notifies you when a job that should be active is not. This list of jobs resides in a CONFIG xx member of SYS1.PARMLIB.

Syntax



where

00 Is the default suffix of the CONFIG xx member.

xx Is the suffix of the CONFIG xx member in SYS1.PARMLIB that is to be used.

The entries in the CONFIG xx member must begin in column 1. The format of the entries is

```
* /NAME=j obname TIME=hhmm- hhmm ACT=command
```

where

j obname Is the jobname (or started task ID for started tasks).

hhmm- hhmm Is the beginning and ending times for the interval that the specified job must be active.

- Valid values for *hh* are 00 to 23.
- Valid values for *mm* are 00 to 59.

command Is an optional field that you can substitute with any OS/390 command. The command executes if the targeted job is not active during the specified time interval. ACT is ignored when the service is not invoked asynchronously. When ACT is specified, *command* cannot be blank.

Example

Assume that warning messages are to be issued if job INVENTORY is not active between 8:00am and 5:00pm. Place the following entry in the CONFIG00 member of SYS1.PARMLIB:

```
* /NAME=I NVENTORY TIME=0800- 1700
```

To display deviation from the CONFIG00 member, type:

```
devi atn 00
AMTCF6I NO DEVIATION FOUND FROM CONFIG00
```

If job INVENTORY is not active during the time specified in the CONFIG00 member, a warning message is displayed, as illustrated below:

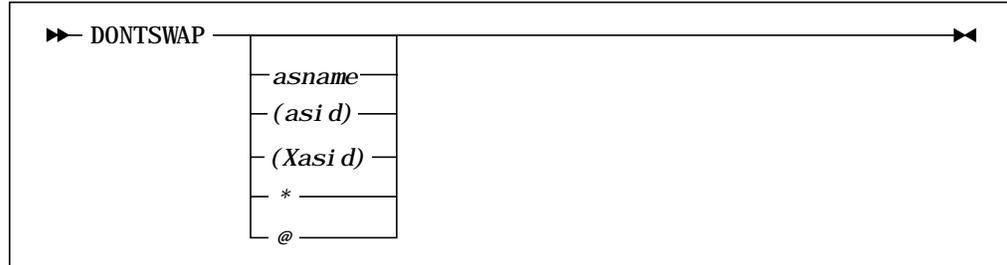
```
devi atn 00
```

```
AMTCF8I WARNING INVENTORY SHOULD BE ACTIVE 0800-1700
```

DONTSWAP

The DONTSWAP (DON) service makes an address space nonswappable.

Syntax



where

- asname* Is the address space name.
- (asi d)* Is the address space identifier in decimal format.
- (Xasi d)* Is the address space identifier in hexadecimal format.
- ** Specifies the last address space entered.
- @* Specifies your own address space.

Note: If you do not specify an address space to be made nonswappable, you are prompted to define an address space.

Example

To make address space INVENTORY nonswappable, type:

```

dontswap inventory
AMTS51I  JOB0205 1INVENTORY STEP1      2PRTY EF (239) 3PGP 2/5
AMTS52I  INVENTORY HAS BEEN SET NONSWAPPABLE 4(01)
    
```

Legend:

1. JES job ID, name, and current step name of the specified address space.
2. External and internal dispatching priority of the address space; the dispatching priority is higher than it was before the DONTSWAP service was issued (however, the condition is temporary and the priority returns to its previous level).
3. Performance group and performance group period.
4. Nonswappability count; the operating system keeps this count for the address space. (If the count is greater than zero, the address space is still nonswappable; when the count reaches zero, the address space is swappable.)

DSNAME

The DSNAME (DSN) service displays information about a specified data set, including:

- Volume serial number
- Device type
- Tracks allocated
- Tracks used
- Number of extents
- Data set organization
- Record format
- Block size
- Logical record length

Syntax

```
▶▶ DSNAME — dsname —▶▶
```

where

dsname Is the data set name.

Examples

To display information about data set SYS1.DUMP05, type:

```
dsname sys1. dump05
AMTD10I DATASET SYS1. DUMP05                               ❶
AMTD13I VOL=DUMPB1; DEVT=3380;   TRACKS ALLOCATED= 2175, USED=  920  ❷
AMTD1EI EXTENTS= 1;                                                       ❸
AMTD14I DSORG=PS;   RECFM=FB;   BLKSIZE= 4160;           LRECL= 4160  ❹
```

To display the volume serial numbers of data set MASTER.TAPE, type:

```
dsname master. tape
AMTD10I   DATASET MASTER. TAPE
AMTD11I   VOLUMES 874008 316002 000381                               ❺
```

Legend:

1. Name of the specified data set.
2. Volume and device type on which the data set resides, as well as the space allocated and used.
3. Number of extents used.
4. RECFM/LRECL/BLKSIZE and organization of the data set.
5. Only volume information is displayed for data sets that reside on tape or for a disk data set that resides on a volume that is not currently mounted.

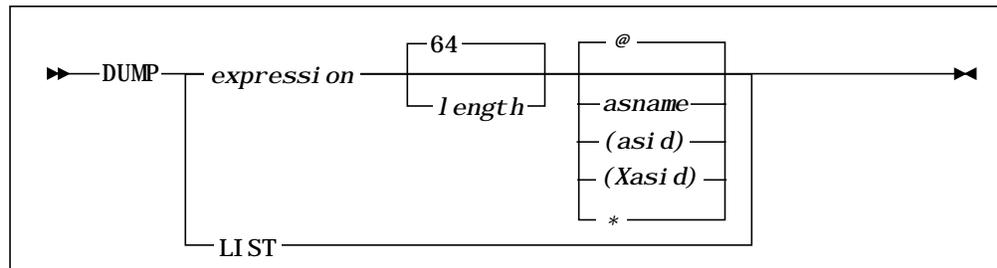
Usage Notes

- Only the volume serial number and data set organization are provided for VSAM data sets.
- For data sets on multiple mounted volumes, the usage data is displayed for each volume.
- A maximum of 20 volume serial numbers can be displayed per command.

DUMP

The DUMP (DU or LI) service displays storage in hexadecimal and character formats.

Syntax



where

expression Specifies the location in memory you want to display. The **expression** parameter has two parts: a memory location and, optionally, any combination of offsets (+ or - hex values) or indirection indicators (% , ? , or >). The memory location can be a hex address, a predefined symbolic name, or an asterisk. The hex address can be from 1 to 8 hex digits. Symbols are defined with the EQUATE service; see [“EQUATE” on page 92](#) for more information. An asterisk represents the start of the area last displayed or the start of a load module located by the LPA service.

When specified, an offset is a 1- to 4-digit hex value preceded by a plus or minus sign. This value indicates the relative distance from the beginning of the location defined by the first part of **expression** to the area in storage to be displayed.

An indirection indicator specifies that the address contained in the full word at the location defined by the first part of **expression** becomes the address to be used next. You can use the following indirection indicators:

- % Indicates the address contained in the word located by the first part of **expression** is treated as a 24-bit address and that this address replaces the first part of **expression** before continuing.
- ? Indicates the address contained in the word located by the first part of **expression** is treated as a 31-bit address and that this address replaces the first part of **expression** before continuing.
- > Indicates the address contained in the three bytes located by the first part of **expression** is treated as an SWA address token and that this token replaces the first part of **expression** before continuing.

length Is the length, in bytes, of the storage area to be dumped. The default length is 64.

@ Specifies your own address space; the default.

<i>asname</i>	Is the address space containing the storage to be displayed.
(<i>asi d</i>)	Is the address space identifier in decimal format.
(<i>Xasi d</i>)	Is the address space identifier in hexadecimal format.
*	When specified as an address space name, indicates the last address space entered.
LIST	Displays the predefined and user-defined symbols, along with their respective definitions, that can be used with DUMP. (See “EQUATE” on page 92 for information on defining symbols.)

Examples

To display storage at hex address 10 in your address space, type:

```

dump 10
      ① ②                ③                ④
AMTC12I 00000010 08 00008400 00000000 FF060080 80000000 *.....*
AMTC12I 00000020      FFB5000A 4F063A02 FFA50008 0E087888 *.....*
AMTC12I 00000030      0000FF00 00000000 FE040130 80000AF6 *.....6*
AMTC12I 00000040      00084DB8 0C800000 00003D18 00008400 *.....*
    
```

Legend:

1. Display address.
2. Virtual storage key (first digit) and fetch-protect bit (high order bit of second digit); the remaining three bits are always zero.
3. Sixteen-byte hex dump per line.
4. Character representation of hex dump.

When DUMP is typed without any operands, it displays a list of all of the symbols created by the user (by way of the EQUATE service) or on behalf of the user (by the MEMSCAN and TCB services).

```

dump
AMTC1HI USER DEFINED SYMBOLS ARE:
AMTC1GI @TCB001 7FD968
AMTC1GI @TCB001 7CAD90
AMTC1GI @TCB003 7CA930
AMTC1FI @TCB004 7BCE88
AMTC1GI @TCB005 7CA5F0
AMTC1GI @TCB006 7EE6B0
AMTC1GI @TCB007 7EEA40
AMTC1G1 @TCB008 7EE468
AMTC1GI @TCB009 7EE150
    
```

To display the symbols that can be used with the DUMP service, type:

```
dump list

AMTC1HI USER DEFINED DUMP SERVICE LABELS ARE:
AMTC1GI @PWVT      00008C88
AMTC1GI @AVT      00009000
AMTC1GI @BBCT     @AVT+58?
AMTC1FI PREDEFINED DUMP SERVICE CONTROL BLOCK LABELS ARE:
AMTC1GI @ACB      10%+100?+14?+18?
AMTC1GI @AMCBS    10%+100?
AMTC1GI @ASCB     224%
AMTC1GI @ASMVT    10%+2C0?
AMTC1GI @JSTCB    224%+6C%+8%+7C%
```

You can use a combination of predefined symbols and indirection indicators to display the CDEs for the job pack area (JPA) for an address space. For example, to locate the beginning of the chain and display the first CDE for address space JES2, type:

```
dump @jstcb+2c? 32 jes2

AMTC12I 007E2000 00 007E2040 00000000 C9C5C6C3 D5C1D4E2 *... .. I EFCNAMS*
AMTC12I 007E2010    800F2340 007E20E8 000210FC 31224000 *... .. Y..... *
```

The +2c? in the command displays the 31-bit address located at +2C from the beginning of the area.

To display the next CDE in the chain, type:

```
dump *+? 32 *

AMTC12I 007E2040 00 007FD580 00000000 C9C5C6E5 C8F14040 *.. N. .... I EJVH1  *
AMTC12I 007E2050    863755C0 007FF060 000210FC 31224000 *..... 0-..... *
```

The first asterisk in the command represents the beginning of the area previously displayed and the second asterisk indicates that the target address space is the same as in the previous command.

To display the next CDE in the chain, simply repeat the previous command.

```
dump *+? 32 *

AMTC12I 007FD580 00 007FD5C0 00000000 C9C5C1E5 D4F7F0F3 *.. N. .... I EAVM703*
AMTC12I 007FD590    86217D88 007FD5E0 000110FC 31224000 *..... N..... *
```

You can repeat this command until all CDEs in the chain have been displayed.

Usage Notes

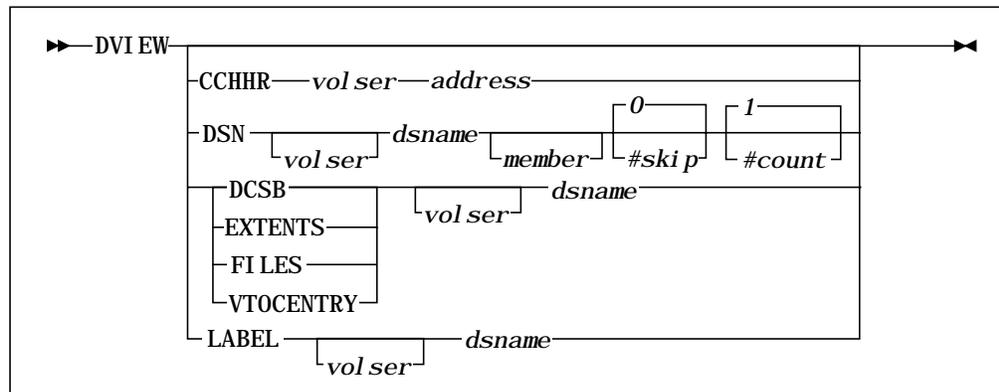
- The DUMP service operates in SRB mode when displaying data. Therefore, PSATOLD (task control block -TCB old pointer of the prefixed saved area) always contains zero and cannot be used to locate the current TCB (task control block).
- ZAP sometimes is used immediately following DUMP. The beginning address of the storage area last displayed by DUMP is retained. The ZAP service can refer to this address by using an asterisk (*) for the *hex* location parameter.
- For processor complexes consisting of more than one CPU, you can specify a CPU number. Add /*c* to the end of the value specified for *expression*, where *c* is the number of an available CPU.
- LIST or LI is a valid alias for the DUMP service.
- If you omit the *length* parameter on the COMMAND line, you can specify the *asname* parameter in its place; for example, DUMP @AVT I NVENTORY, where I NVENTORY is an address space name.

DVIEW

The DVIEW (DV) service displays data blocks from any online DASD volume. DVIEW assists in locating the data blocks to be displayed by searching the VTOC and PDS directories for a specified data set name. You also can use DVIEW to display:

- Extents of a data set
- DSCB of a data set
- List of data sets on the volume
- A volume's VTOC
- A volume's label

Syntax



where

CCHHR Displays a data block using its cylinder, track, and record number.

vol ser Specifies the volume serial number of the DASD volume from which the data block is displayed. If this field is not specified, *vol ser* is obtained from the catalog.

address Is a required field that specifies the cylinder, head, and record of the data block to be displayed; the format of the operand is a 10-digit hex value, cccchhhrr.

DSN Displays the data blocks requested from a specified data set or member.

dsname Is a required field that specifies the data set name to be displayed; use the special name *FMT4 to specify the VTOC.

Note: Enclose *FMT4 in single quotes ('*FMT4') when specifying this field from the SYSPROG Services selection panel.

member Is an optional part of the dsname that specifies the member of a partitioned data set from which the data blocks are displayed.

<i>#skip</i>	Is an optional field that specifies the number of data blocks that are to be skipped; the default value is zero (an EOF block does not stop the operation and is counted as a block).
<i>#count</i>	Is an optional field that specifies the number of data blocks that are to be displayed; the default value is one (an EOF block stops the operation).
DSCB	Displays the Format 1 DSCB of a specified data set.
EXTENTS	Displays an extent list for the specified data set.
FILES	Lists the file names in the VTOC that begin with a partial data set name.
VTOCENTRY	Displays the VTOC entry for the specified data set.
LABEL	Displays the label data block of a volume.

Note: You can abbreviate the first operand to a nonambiguous character. For example, you can abbreviate CCHHR to C because no other operands in the DVIEW syntax diagram begin with that letter. However, you cannot abbreviate DSCB to D or even DS because these abbreviations conflict with the DSN operand; instead, abbreviate DSCB to DSC.

Examples

The following examples illustrate the ways to invoke the DVIEW service.

CCHHR Example

To display the data block found on volume PMG005 at cylinder 40, track 13, record 26, type:

```

dview cchhr pmg005 0028000d1a

AMFVD08I

AMFVD02I  DISK VIEW

AMFVD03I  VOL - PMG005                BLOCK CCHHR - 0028000D1A
AMFVD06I  CSW - 0008E248 0C00E7EF  BLKSIZE - 6160
AMFVD11I  ECB - 4102C78C                SENSE - 0000
AMFVD08I
AMFVD07I  0000  40404040  40404040  40E2D7C1  C3C540F2  *          SPACE 2*
AMFVD07I  0010  40404040  40404040  40404040  40404040  *          *
AMFVD07I  0020  40404040  40404040  40404040  40404040  *          *
AMFVD07I  0030  40404040  40404040  40404040  40404040  *          *
AMFVD07I  0040  40404040  40404040  F0F1F5F5  F0F0F0F0  *          01550000*
AMFVD07I  0050  40404040  40404040  40D34040  404040D9  *          L      R*
AMFVD07I  0060  F16BC9D5  E3C5D9E5  C1D34040  40404040  *1. INTERVAL *
AMFVD07I  0070  40404040  C9D5E3C5  D9E5C1D3  40C9D540  *          INTERVAL IN *
AMFVD07I  0080  C8E4D5C4  D9C5E3C8  E2404040  40404040  *HUNDREDTHS *
AMFVD07I  0090  40404040  40404040  F0F1F5F6  F0F0F0F0  *          01560000*
AMFVD07I  00A0  40404040  40404040  40D34040  404040D9  *          L      R*
    
```

DSN Example

To display the first data block of data set USER.DATA on volume PMG005, type:

```

dview dsn pmg005 user. data

AMTVD08I
AMTVD02I DISK VIEW
AMTVD03I VOL - PMG005          BLOCK CCHHR - 01D0000401
AMTVD06I CSW - 0008D248 0C00FEF7 BLKSI ZE - 264
AMTVD11I ECB - 4102C78C          SENSE - 0000
AMTVD08I
AMTVD07I 0000 C1D5C1D3 E8E9D9F1 00FEC1C4 D9C4E2E2 *ANALYZR1..ADRSS*
AMTVD07I 0010 E440000E 050F0100 00000087 012F0087 *U.....g...g*
AMTVD07I 0020 012F1302 00160016 0000C3D4 C6F44040 *.....CMF4 *
AMTVD07I 0030 40404040 C1D4C1E2 D7E9C1D7 001B070F *  AMASPZAP....*
AMTVD07I 0040 01030000 0087012F 0087190F 1105000E *.....g...g.....*
AMTVD07I 0050 000A0009 C3D4C6F4 40404040 4040C1D4 *...CMF4      AM*
AMTVD07I 0060 C2D3C9E2 E3400002 050F0100 00000087 *BLIST.....g*
AMTVD07I 0070 012F0087 012F1302 000A000A 0000C3D4 *...g.....CM*
AMTVD07I 0080 C6F44040 40404040 C1D4C4D7 D9C4D4D7 *F4      AMDPRDMP*
AMTVD07I 0090 0014030F 01010000 0087012F 0087069F *.....g...g...*
AMTVD07I 00A0 10210012 00120004 C3D4C6F4 C2404040 *.....CMF4B *
AMTVD07I 00B0 4040C1D5 C1D3E8E9 C5D9000C 090F0100 *  ANALYZER.....*
AMTVD07I 00C0 00000087 012F0087 012F1302 01340134 *...g...g.....*
AMTVD07I 00D0 0000C3D4 C6F44040 40404040 C1D5C1D3 *..CMF4      ANAL*
AMTVD07I 00E0 E8E9D9F1 000E010F 01000000 0087012F *YZR1.....g...*
AMTVD07I 00F0 0087012F 1302001F 001F0000 C3D4C6F4 *.g.....CMF4*
AMTVD07I 0100 40404040 40400000 * .. *
    
```

DSCB Example

To display the Format 4 DSCB on volume PMG005, type:

```

dview dsch pmg005 *fmt4

AMTVD08I
AMTVD02I DISK VIEW
AMTVD03I VOL - PMG005          BLOCK CCHHR - 00B4000001
AMTVD06I CSW - 0008C248 0C00FF73 BLKSI ZE - 140
AMTVD11I ECB - 4102C78C          SENSE - 0000
AMTVD08I
AMTVD07I 0000 04040404 04040404 04040404 04040404 *.....*
AMTVD07I 0010 04040404 04040404 04040404 04040404 *.....*
AMTVD07I 0020 04040404 04040404 04040404 F400B600 *.....4...*
AMTVD07I 0030 1D2F0FB8 022B0000 00968901 00000230 *.....oi....*
AMTVD07I 0040 001E4B36 010B5209 02002F24 00000000 *.....*
AMTVD07I 0050 00000000 00000000 00000000 00000000 *.....*
AMTVD07I 0060 00000000 00000000 00010000 B4000000 *.....*
AMTVD07I 0070 B6001D00 00000000 00000000 00000000 *.....*
AMTVD07I 0080 00000000 00000000 00000000 *.....*
    
```

EXTENTS Example

To display an extent list of the data set USER.DATA on volume PMG005, type:

```

dview extents png005 user.data

AMTVD08I
AMTVD25I DSN - USER. DATA
AMTVD08I
AMTVD26I FLAGS START CCHH END CCHH
AMTVD27I 0100 01D00004 01D2000F
AMTVD27I 0101 000C000F 000C0010
AMTVD08I
    
```

FILES Example

To list the names of all data sets on volume PMG005 that begin with USER, type:

```

dview files png005 user

AMTVD14I USER. TEXT. CNTL DSCB - 00B4000016
AMTVD14I USER. TEXT. CLI B DSCB - 00B4000221
AMTVD14I USER. TEXT. TEXT DSCB - 00B4000303
    
```

VTOCENTRY Example

To display the VTOC entry for the data set USER.DATA on volume PMG005, type:

```

dview vtocentry png005 user.data

AMTVD08I

AMTVD02I DISK VIEW

AMTVD03I VOL - BAB319 BLOCK CCHHR - 0685000B1F
AMTVD06I CSW - 00083288 0C00FF73 BLKSIZE - 140
AMTVD11I ECB - 411116D0 SENSE - 0000
AMTVD08I
AMTVD07I 0000 D4E9C2F1 4BC2D6D6 D2D4C1E2 E34BD4C1 *TS01. USER. DATA *
AMTVD07I 0010 E3E34040 40404040 40404040 40404040 * *
AMTVD07I 0020 40404040 40404040 40404040 F1404040 * 1 *
AMTVD07I 0030 5D016C00 015D00A0 00000002 0000D4E9 *.....TS*
AMTVD07I 0040 C2F14040 4040D4E9 C2F1405E 00030000 *01 TS01 .....*
AMTVD07I 0050 00000200 90000A64 00850000 00808000 *.....e.....*
AMTVD07I 0060 00280043 103762F1 00010001 6D000001 *.....1.....*
AMTVD07I 0070 71000501 01057800 0E057900 05000000 *.....*
AMTVD07I 0080 00000000 00000000 00000000 *.....*
    
```

LABEL Example

To display the label data block of volume PMG005, type:

```

dview label pmg005

AMTVD08I

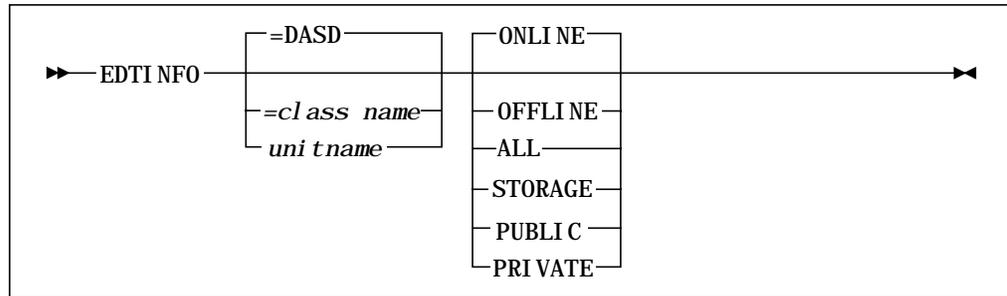
AMTVD02I DISK VIEW

AMTVD03I VOL - PMG005          BLOCK CCHHR - 0000000003
AMTVD06I CSW - 0008D248 0C00FFAB BLKSIZE -      84
AMTVD11I ECB - 4102C78C          SENSE - 0000
AMTVD08I
AMTVD07I 0000 E5D6D3F1 E5D6D3F1 D7D4C7F0 F0F54000 *VOL1VOL1PMG005.*
AMTVD07I 0010 B4000001 40404040 40404040 40404040 *... *
AMTVD07I 0020 40404040 40404040 40C2D6D6 D3C54040 *      CORPE *
AMTVD07I 0030 40404040 40404040 40404040 40404040 *      *
AMTVD07I 0040 40404040 40404040 40404040 40404040 *      *
AMTVD07I 0050 40404040                *      *
```

EDTINFO

The EDTINFO service provides information about unitnames in the current Eligible Device Table (EDT). Unitnames are subdivided into *generic* and *esoteric*. Generic names are IBM device type names such as 3380, 3390, or 3400-6. Esoteric names are user defined; for example, SYSDA, TAPE, and CART.

Syntax



where

- =class name** Displays a list of the generic and esoteric unitnames for the specified class name. You must prefix the class name with an equal (=) sign. The default is =DASD. Supported class names are: DASD, TAPE, UREC, COMM, CTC, CHAR, and DISP.
- unitname** Displays the device number, device type, volser, and the use status of the specified unitname.
- ONLINE** Limits the display to online devices; the default.
- OFFLINE** Limits the display to offline devices.
- ALL** Displays information for all devices.
- STORAGE** For DASD devices, limits the display to devices containing volumes with a use attribute of STORAGE. This parameter is ignored for other device types.
- PUBLIC** For DASD devices, limits the display to devices containing volumes with a use attribute of PUBLIC. This parameter is ignored for other device types.
- PRIVATE** For DASD devices, limits the display to devices containing volumes with a use attribute of PRIVATE. This parameter is ignored for other device types.

Examples

To display the generic and esoteric unitnames for device class DASD, type:

```

edti info
AMTED1I          DEVI CE          ①
AMTED2I  UNI TNAME  COUNT  FOR CLASS=DASD
          ②
AMTED3I  2305-2      1
AMTED3I  3390        144
AMTED3I  3330        1
AMTED3I  ALLDA      608
AMTED3I  SYSTS0     384          ③
AMTED3I  SYSVI O    1  VI O ELI GI BLE
AMTED3I  TSGDA      16
AMTED3I  VI O        1  VI O ELI GI BLE
AMTED3I  VI O3390    1  VI O ELI GI BLE
AMTED3I  SYSALLDA   660

```

Legend:

1. Device class being displayed.
2. Number of devices defined for the respective generic or esoteric unitname.
3. Indicates that the unitname is VIO eligible.

To display information about the online devices for the esoteric unitname CART, type:

```

edti info cart
AMTED4I  16 DEVICES FOR ESOTERIC UNITNAME CART. DISPLAYING ONLINE ONLY.
          ③ ④ ⑤
AMTED5I  DVN DEV TYPE VOLSER  STATUS
AMTED6I  480 3480    CC7273
AMTED6I  488 3480    501697
AMTED6I  489 3480    508881

```

Legend:

1. Indicates that the unitname is esoteric.
2. Indicates that only online devices are displayed.
3. Device number.
4. Device type. (A unitname may contain devices with different device types.)
5. Device volume serial number. VOLSER is applicable only to DASD and TAPE devices.

To display information about all the devices for the generic unitname 3400-6, type:

```

edtinfo 3400-6 all
AMTED4I      31 DEVICES FOR GENERIC UNITNAME 3400-6. DISPLAYING ALL. ❶
AMTED5I      DVN DEV TYPE VOLSER STATUS ❷
AMTED6I      180 3400-6 OFFLINE
AMTED6I      181 3400-6 OFFLINE
AMTED6I      182 3400-6 OFFLINE
AMTED6I      183 3400-6 ONLINE
AMTED6I      184 3400-6 OFFLINE
AMTED6I      185 3400-6 OFFLINE

```

Legend:

1. Indicates that both online and offline devices are displayed.
2. Use status of the device.

To display the total number of generic unitname 3390, type:

```

edtinfo, 3390
AMTED4I      160 DEVICES for Generic unitname 3390. Displaying online only. ❶
AMTED5I      DVN Dev Type VOLSER Status ❷
AMTED6I      2A0 3390 HSM301 Private
AMTED6I      2A2 3390 SYSS1C Private
AMTED6I      2A3 3390 ES522D Private
AMTED6I      2A4 3390 STRA03 Storage
AMTED6I      2A7 3390 SP52EM Private

```

Legend:

1. Indicates that only online devices are displayed.
2. Use attributes and status of the device.

To display only those devices with the generic name 3390 that are in storage status, type:

```

edtinfo, 3390, storage

AMTED4I    160 Devices for Generic unitname 3390. Storage only. ❶
                ❷
AMTED5I    DVN Dev Type VOLSER  Status
AMTED6I    2A4 3390     STRA03 Storage
AMTED6I    323 3390     FAT902 Storage
AMTED6I    2A5 3390     STRBC1 Storage
AMTED6I    309 3390     BAB318 Storage
AMTED6I    320 3390     FAT900 Storage
AMTED6I    322 3390     FAT901 Storage
AMTED6I    D27 3390     TSG306 Storage
AMTED6I    D2A 3390     TSG311 Storage
AMTED6I    D31 3390     EMP301 Storage

```

Legend:

1. Indicates that only storage devices are displayed.
2. Use attributes and status of the device.

Usage Notes

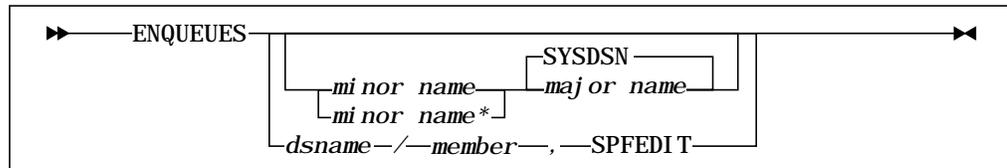
- The class name may be abbreviated to the shortest unique value; for example, =T or =TA (for TAPE) and =U or =UR (for UREC).
- Unlike class names, unitnames cannot be abbreviated.
- Devices may be (and often are) included in multiple unitnames.
- For DASD devices, the use status, PRIVATE, PUBLIC, or STORAGE, is displayed.

ENQUEUEES

The ENQUEUEES (ENQ) service displays current enqueue conflicts or requested system enqueue information. An enqueue conflict exists when one or more jobs are waiting on a resource held by another job. This service lets you determine the cause of reduced system performance, the name and system ID for jobs holding a resource, and the jobs waiting for the same resource.

When you type the ENQUEUEES command without operands, information about any current enqueue conflicts is displayed. The ENQUEUEES service also displays enqueue information for a specified minor name and major name (R name and Q name, respectively), or all enqueue information for a specified major name (Q name). The default major name is SYSDSN. Therefore, you can display data set enqueues simply by typing the minor name (data set name).

Syntax



where

- minor name* Is the R name used by ENQUEUEE.
- minor name** Indicates a partial minor name. Partial names may be specified using a wildcard character (asterisk).
- major name* Is the Q name used by ENQUEUEE; SYSDSN is the default.
- dsname* Is a data set name.
- member* Is a member name.
- SPFEDIT Is the major name used by ISPF EDIT.

Examples

The following examples demonstrate how the ENQUEUEES service presents information on enqueue conflicts and resource ownership.

Conflict Information Example

An enqueue conflict exists when one or more jobs wait for a resource held by another job. The ENQUEUEES service displays the name and system ID of jobs holding a resource and the jobs waiting for the same resource. Use this information to determine the source of the performance delay in your system.

To display all current enqueue conflicts, type:

enqueues														
AMTQ1PI	SYSTEM	(LOCAL)	Q=SYSZVVDS	R=CATALOG.	ICFMCAT.	SYSC								
AMTQ1QI	SYSD	JOBNAME	ASID	STAT	TYP	TIME	14:12:19							
AMTQ1RI	SYSB	QA7A	(0087)	OWNS	EXC									
AMTQ1RI	SYSA	FLN1	(0096)	WAIT	EXC									
AMTQ1RI	SYSB	QA7	(0352)	WAIT	EXC									
AMTQ1LI														
AMTQ1PI	SYSTEM	(LOCAL)	Q=SYSI GGV2	R=ICFUCAT.	VTSG304									
AMTQ1QI	SYSD	JOBNAME	ASID	STAT	TYP	TIME	14:12:19							
AMTQ1RI	SYSB	ARG1	(0162)	WAIT	EXC									
AMTQ1RI	SYSC	MPP1	(0121)	OWNS	SHR	RES=003	PEND	291	TSG304	NR				

Legend:

1. Scope of enqueue. The possible scopes are as follows:
 - SYSTEMS
 - SYSTEM
 - STEP
2. Whether the resource is global or local.
3. Major name (Q=).
4. Minor name (R=).
5. System ID for the system executing the task that is holding or waiting for the resource.
6. Jobname, TSO user ID, or started task ID for the address space containing the task that is holding or waiting for the resource. The jobname is not available in some situations.
7. ID for the address space holding or waiting for the resource.
8. Current status (OWNS or WAIT), indicating that the task holds (OWNS) the resource or is waiting (WAIT) for the resource.
9. Type of enqueue: EXC for exclusive, SHR for shared.
10. Indicates that a reserve is associated with the enqueue. The reserve count is also displayed unless it is zero, in which case the equal sign is also omitted.
11. If present, indicates that the reserve is pending, which means that the task is waiting to reserve the devices. Generally, the device is reserved by another system.
12. Device number in hexadecimal.
13. Volume serial number.
14. If present, indicates that the device is not ready.

Resource Ownership Information Examples

Typing ENQUEUES followed by the minor name (R name) and major name (Q name) displays the job(s) that holds the specified resource and information related to its use of the resource. A partial minor name displays all enqueues with minor names that begin with the specified characters for the requested major name. An asterisk typed as the last character indicates a partial minor name. SYSDSN is the default major name.

To display all enqueues for data sets starting with SYS1.L, type:

```
enqueues sys1.l*
```

```
AMTQ1P1 SYSTEM (LOCAL) Q=SYSDSN R=SYS1.LPALIB
AMTQ1QI SYSID JOBNAME ASID STAT TYP TIME 16:05:20
AMTQ1RI SYSB MEE2 (0192) OWNS SHR
AMTQ1LI
AMTQ1P1 SYSTEM (LOCAL) Q=SYSDSN R=SYS1.LINKLIB
AMTQ1QI SYSID JOBNAME ASID STAT TYP TIME 14:12:19
AMTQ1RI SYSB LLA (0006) OWNS SHR
```

Besides SYSDSN, another common major name is SPFEDIT. ISPF EDIT uses the major name SPFEDIT and a 52-character minor name consisting of the data set name (a 44-character field padded with blanks), followed by the member name.

The ENQUEUES service provides special syntax to allow you to display these ISPF EDIT enqueues for specific members. Type the data set name (dsn), followed by a slash (/), followed by the member name, a comma (or blank), and the major name SPFEDIT.

For example, to determine the jobnames editing member IEASYS00 in SYS1.PARMLIB, type:

```
enqueues sys1.parmlib/ieasys00,spfedit
```

```
AMTQ1P1 SYSTEM (LOCAL) Q=SPFEDIT R=SYS1.PARMLIB IEASYS00
AMTQ1QI SYSID JOBNAME ASID STAT TYP TIME 14:12:19
AMTQ1RI SYSB CIR3 (0183) OWNS EXC
```

In each of these examples, the header line is followed by two or more information lines. The ENQUEUES service displays one line for each task that issued an ENQ or RESERVE for the resource.

To display all jobs using all resources for a major name, type an asterisk (or a comma to indicate no minor name), followed by the major name, as illustrated below:

```
enqueues *,syszvvds
```

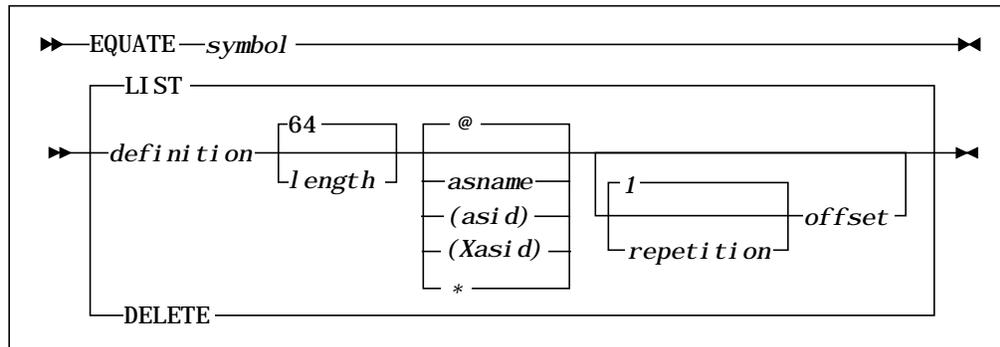
```
AMTQ1P1 SYSTEM (LOCAL) Q=SYSZVVDS R=CATALOG.ICFMCAT.SYSC
AMTQ1QI SYSID JOBNAME ASID STAT TYP TIME 14:12:19
AMTQ1RI SYSB QA7A (0087) OWNS EXC
AMTQ1RI SYSB FLN1 (0096) WAIT EXC
AMTQ1RI SYSD QA7 (0352) WAIT EXC
```

Note: Specifying a partial minor name may produce a large volume of output.

EQUATE

The EQUATE (EQ) service assigns a symbolic name to an address expression. The DUMP service permits symbols to be used in place of address expressions. EQUATE lets you define, redefine, and display symbol definitions.

Syntax



where

- symbol* Is the symbol to be defined or displayed. The symbol must begin with @ and can be followed by one to seven alphanumeric characters.
- LIST Displays the definition of the symbol, if previously defined. The default.
- definition* Is an expression that defines the symbol. This can begin either with a previously defined symbol (including the one being defined) or a hexadecimal address, optionally followed by offset values (+ or -) and/or indirection indicators (% or ?). For example, the start of the CVT prefix can be defined as 10%-100.
- length* Specifies a length between 1 and 4096 bytes that is associated with the symbol; the default length is 64.
- @ Specifies your own address space. The default.
- asname* Specifies the address space name associated with the symbol.
- (asid)* Is the address space identifier in decimal format.
- (Xasid)* Is the address space identifier in hexadecimal format.
- * Specifies the last address space typed.
- repetition* Specifies the number of control blocks, between 1 and 99, that are displayed; the default is 1. When this parameter is present, the DUMP service assumes that the symbol describes a chain of control blocks to be followed until the end of the chain is reached or the repetition count is exhausted. The pointer to the next control block is defined by the *offset* parameter.

Note: When you specify the *repetition* parameter, you must also specify the *offset* parameter.

<i>offset</i>	Is a 1- to 4-digit hexadecimal value preceded by a plus (+) or minus (-) sign indicating the relative location of the pointer that contains the address of the next storage area displayed. The pointer must be a full word. A pointer value of 0 terminates the display. Prefixing <i>offset</i> with a percent symbol (%) indicates the pointer is a 24-bit address; prefixing <i>offset</i> with a question mark (?) indicates the pointer is a 31-bit address. The default is a 31-bit address.
DELETE	Deletes the symbol.

Examples

To display the definition of symbol @JSCB, type:

```
equate @jscb
AMTEQ7I @JSCB=224%+6C%+8%+7C%+B4%, 380
```

To define symbol @A as symbol @JSCB + 2% and give it a length of 64 bytes, type:

```
equate @a @jscb+2% 64
AMTEQ9I SYMBOL DEFINED
```

To delete symbol @A, type:

```
equate @a delete
AMTEQ4I @A DELETED
```

ESCLASS

The ESCLASS (ESC) service provides the ability to:

1. Display a list of all service classes in the current service policy and their description.

This display is produced when the ESCLASS service is executed without any operands.

2. Display the service class for a specific address space and service-related information.

This display is produced when a jobname or ASID is specified as the first and only parameter. The ASID must be enclosed in parentheses and is assumed to be a decimal value unless preceded by an *x*. I.e., (x24) = (36).

3. Change the service class for an address space.

This action is taken when a valid jobname (or ASID) is specified as the first parameter and a valid service class is specified as the second parameter.

Example: ESCLASS, (x24), newclass will change the service class for the job in the address space with an ASID of decimal 36 (hex 24) to newclass.

4. Quiesce an address space by specifying a valid jobname or ASID as the first parameter and Q or QUIESCE as the second parameter.

Example: ESCLASS job1, Q will quiesce job1.

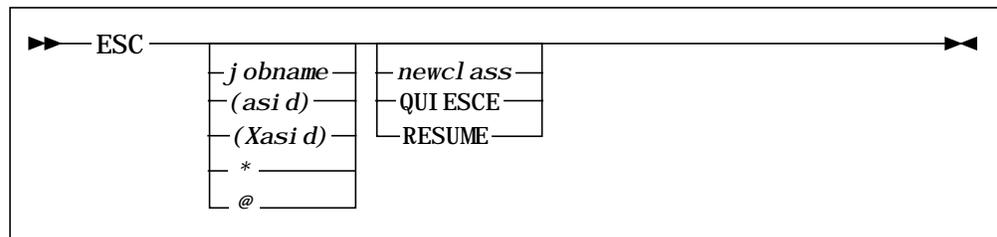
5. Resume (restart) a job that has been quiesced and cause the service class to be set according to the current service policy.

Specify the name of a job or an ASID as the first parameter and the keyword RESUME as the second parameter.

Note: Resuming a job causes the appropriate service class in the current service policy to be applied to the specified job. If you are resetting the job to reverse a QUIESCE, then check the service class that is displayed to ensure that it is the desired service class. If not, use the ESCLASS service to change the service class. An example of resuming an address space:

```
ESCLASS, job1, RESUME
```

Syntax



where:

jobname	Is the job name.
(asid)	Is the address space identifier in decimal format.

(Xasid)	Is the address space identifier in hexadecimal format.
*	Specifies the last address space entered.
@	Specifies your own address space.
newclass	Is the new service class name to be applied to the address space.
QUIESCE	Indicates that the address space is to be quiesced.
RESUME	Indicates that the service class for the specified address space is to be set according to the current service policy.

Example 1

To display a list of all service classes in the current service policy, type:

```

esc

AMTES4I Policy name:  BBPOL002
AMTES4I Policy desc:  Boole & Babbage Test Policy
AMTES4I Activated by: BTSSEC3
AMTES4I From:        SYSC
AMTES4I On:          Tuesday, December 15, 1998 at 6:17:06 AM
AMTES0I
AMTES5I Class      Description
AMTES6I -----
AMTES7I APPCHOT    APPC Hot Transactions
AMTES7I APPCNRM    APPC Normal Transactions
AMTES7I BATHOT     Batch Hot Jobs
AMTES7I BATNRM     Batch Normal Jobs
AMTES7I BATPROD    Batch Production Jobs
AMTES7I CICSHOT    CICS Hot Transactions
AMTES7I CICSNRM    CICS Normal Transactions
AMTES7I CICST1     T1** TRANS for J. Barnard
AMTES7I CICST2     T2** TRANS for J. Barnard
AMTES7I CICST3     T3** TRANS for J. Barnard
AMTES7I CICST4     T4** Trans for J. Barnard
AMTES7I CICST5     T5** TRANS for J. Barnard
AMTES7I COMPOSIT   composite service class
AMTES7I GRS        GRS Service Class
AMTES7I IMSHOT     IMS Hot Transactions
AMTES7I IMSNRM     IMS Normal Transactions
AMTES7I OMVSNRM    Open MVS Normal Transactions
AMTES7I RMF        rmf extractor service class
AMTES7I RMFGAT     RMF III Gatherer Service Class
AMTES7I SERVERS    Service class for Servers
AMTES7I STCLOW     Low Priority STC' s
AMTES7I STCNONE    stc no service
AMTES7I STCNRM     Normal STC' s
AMTES7I STCPAS     PAS STC' s
AMTES7I STCPROD    Production STC' s
AMTES7I STCSYS     System STC' s
AMTES7I SWAPOUT    no service
AMTES7I TSOAVG     Normal TSO Users
AMTES7I TSOBBV     TSO BBV Users
AMTES7I TSODEMO    TSO DEMO Users
AMTES7I TSONRM     Normal TSO Users
AMTES7I TSONRM1    Normal TSO Users 2
AMTES7I TSOQ2      q/a TSO Users
AMTES7I TSOTEST    MMR Test TSO Service Class
AMTES7I SYSTEM     HIGH PRIORITY SYSTEM WORK
AMTES7I SYSSTC     STARTED TASK DEFAULT
AMTES7I SYSOTHER   UNCLASSIFIED WORK

```

Example 2

To display the service class and related service information for a specific address space, type:

```

esc, xtsthpas
AMTES1I Job name:          XTSTHPAS
AMTES1I Service class:    STCPAS
AMTES1I Description:     PAS STC' s
AMTES1I Resource group:  PASSTC
AMTES1I Current period:  1
AMTES2I Importance lvl:  3
AMTES2I Period type:     Velocity goal
AMTES2I Goal :           60%

```

Example 3

To change the service class for a specified address space and display the information, type:

```

esc, xtsthpas, stcnrm
AMTES3I Service class changed for address space XTSTHPAS
AMTES0I
AMTES1I Job name:          XTSTHPAS
AMTES1I Service class:    STCNRM
AMTES1I Description:     Normal STC' s
AMTES1I Current period:  1
AMTES2I Importance lvl:  4
AMTES2I Period type:     Velocity goal
AMTES2I Goal :           30%

```

Example 4

To quiesce a specified address, type:

```

esc, xtsthpas, q
AMTES9I Address space XTSTHPAS has been quiesced.

```

Example 5

To resume (restart) a specified address space and display the information, type:

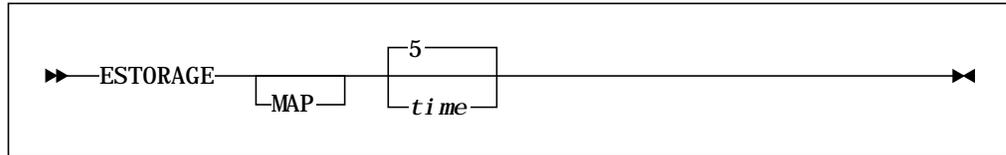
```
esc, xtsthpas, resume
```

```
AMTES8I Resume successful for address space XTSTHPAS
AMTES0I
AMTES1I Job name:          XTSTHPAS
AMTES1I Service class:    STCPAS
AMTES1I Description:      PAS STC' s
AMTES1I Resource group:   PASSTC
AMTES1I Current period:   1
AMTES2I Importance lvl:   3
AMTES2I Period type:      Velocity goal
AMTES2I Goal:             60%
```

ESTORAGE

The ESTORAGE (ES) service provides information on the use of expanded storage for paging. This service lets system programmers verify that expanded storage (sometimes referred to as extended storage) is being used effectively.

Syntax



where

MAP Provides a detailed display of the expanded storage frames in use by each pageable job in the system.

time Specifies the length in seconds of the sample period; the default sample period is 5 seconds.

Examples

To display information about expanded storage, type:

```

estorage

AMTR30I STATISTICS BEING GATHERED FOR EXPANDED STORAGE DATA
AMTR31I ESF INSTALLED/ONLINE 1024/ 1024          ①
AMTR32I AVAILABLE ESF              768          ②
AMTR33I REAL STORAGE MOVEMENT TO ES 20.00 PAGES/SEC ③
AMTR34I ES MI GRATION TO AUX STORAGE 18.00 PAGES/SEC ④
AMTR35I AVERAGE MI GRATION AGE      40 SEC      ⑤
    
```

Legend:

1. Number of expanded storage frames installed and the number currently online.
2. Number of expanded storage frames currently available for use.
3. Rate of movement of pages from real storage to expanded storage.
4. Rate of movement of pages from expanded storage to auxiliary storage.
5. Average length of time that a page resides in expanded storage before being migrated to auxiliary storage.

To display additional information about expanded storage frames, type:

```

estorage map

AMTR30I STATISTICS BEING GATHERED FOR EXPANDED STORAGE DATA
AMTR31I ESF INSTALLED/ONLINE    1024/   1024
AMTR32I AVAILABLE ESF                768
AMTR33I REAL STORAGE MOVEMENT TO ES 20.00 PAGES/SEC
AMTR34I ES MI GRATION TO AUX STORAGE 18.00 PAGES/SEC
AMTR35I AVERAGE MI GRATION AGE      40 SEC

      ①          ②          ③
AMTR36I  JOBNAME      ASID      ESF
AMTR37I  I NVENTORY   0017      0038
AMTR37I  TESTJOB1    0023      0062

```

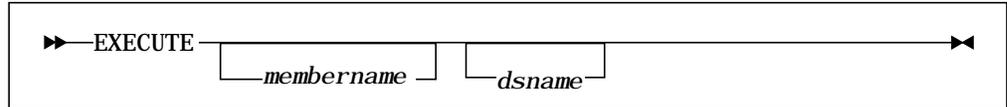
Legend:

1. A line is displayed for each address space that owns expanded storage frames.
2. Address space ID for each job.
3. Number of expanded storage frames currently owned by each job.

EXECUTE

The EXECUTE (EXE) service executes a list of SYSPROG Services, OS/390, or JES commands listed in a member within the specified data set. This service lets you invoke standard sequences of commands. Each SYSPROG Services command in the data set must be prefixed with @ (for example, @ENQ, @REPLI ES).

Syntax



where

membername Is a member within the specified or default data set.

dsname Is the specified data set name. If you do not specify a data set name, EXECUTE will use the previously specified data set.

If there is no previously specified data set, EXECUTE will use the data set allocated to DDNAME BBPARM.

If there is no data set allocated to DDNAME BBPARM, EXECUTE will use the data set allocated to DDNAME LIB.

If there is no data set allocated to DDNAME LIB, MainView for OS/390 will display an error message.

Example

To execute the commands listed in BBPARM library member SHIFT1, type:

```

exe, shi ft1
AMTC21I  COMMAND V 052, ONLINE  EXECUTED FROM CN(01)
AMTC21I  COMMAND V 053, OFFLINE EXECUTED FROM CN(01)
    
```

Legend:

1. Command listed in member SHIFT1 has been executed.

In the following example, the EXECUTE service is used to execute three services contained in member STATUS in the data set VAM3.RES.LIB. The services are preceded by the @ symbol, indicating they are MainView for OS/390 services. The services are executed sequentially with the results displayed after each service name. To execute the contents of the member STATUS in the data set VAM3.RES.LIB, type:

```

exe, status, vam3.res.lib

AMTK11I @RES ❶
AMTQ2PI SYSTEMS (GLOBAL) Q=SYSGGV2 R=ICFMCAT. SYSB ❷
AMTQ2QI SYSID JOBNAME ASID STAT TYP TIME 9:11:08
AMTQ2RI SYSB CATALOG (0023) OWNS SHR CVT
AMTK10I
AMTK11I @ENQ ❶
AMTQ15I NO ENQ CONFLICTS EXIST ❷
AMTK10I
AMTK11I @IO ❶
AMTI11I +*MASTER* Unit 227 PAGD27 IOQ 00FA4E00 Driver-ASM CCHH- 157 3❷
AMTI11I XCFAS Unit 800 IOQ 00FA4C00 Driver-*UNKNOWN
AMTI11I +XCFAS Unit 801 IOQ 00FC1100 Driver-*UNKNOWN
AMTI11I XCFAS Unit 804 IOQ 00FA4900 Driver-*UNKNOWN
AMTI11I +XCFAS Unit 805 IOQ 00FA4F80 Driver-*UNKNOWN
AMTK10I

```

Legend:

1. Name of the service to be executed.
2. Results of the executed service.

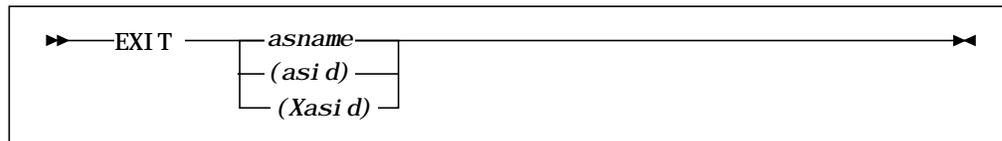
Usage Notes

- The OS/390 and JES2 commands are executed using the COMMAND service. Each OS/390 command that is executed is logged at the issuing console.
- The data set name specified (or the default data set) is not saved across sessions. You must specify a data set name, or accept the defaults, each time you restart MainView for OS/390.
- You can display the previously entered data set name by typing EXECUTE without any parameters.

EXIT

The EXIT (EXI) service terminates an OS/390 address space that does not respond to CANCEL or STOP commands from the system. This service also recovers system resources (for example, initiator, devices, data sets) by forcing an address space to terminate.

Syntax



where

<i>asname</i>	Is the address space name.
<i>(asi d)</i>	Is the address space identifier in decimal format.
<i>(Xasi d)</i>	Is the address space identifier in hexadecimal format.

Example

To terminate address space INVENTORY, type:

```

exit inventory
AMTE22I  ADDRESS SPACE WILL BE TERMINATED BY RTM
  
```

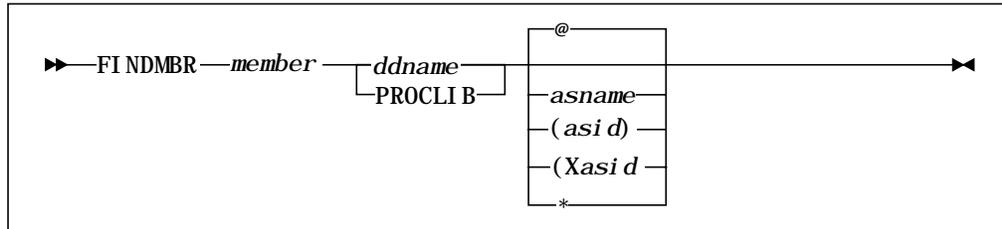
Usage Note

Use the EXIT service only as a last effort to avoid an IPL. The EXIT service issues a CALLRTM to pass the job to the Recovery Termination Manager (RTM) for termination processing and does nothing to clean up after the target address space. As a result, CSA that was allocated by the address space may not be freed. Any recovery done is performed by RTM.

FINDMBR

The FINDMBR (FI) service identifies the names of all libraries in a specified DD name that contain a specified member.

Syntax



where

<i>member</i>	Is the member name.
<i>ddname</i>	Is the DD name. When aliases are used, the true member name appears in the output.
PROCLIB	Specifies that all PROC <i>nn</i> (00 through 99) DD names for JES2 or JES3 are to be searched for a member name. If no PROC <i>nn</i> DD names are allocated, the PROCLIB DD name is searched, if allocated.
@	Specifies your own address space; the default.
<i>aname</i>	Is the address space name.
(<i>asid</i>)	Is the address space identifier in decimal format.
(<i>Xasid</i>)	Is the address space identifier in hexadecimal format.
*	Specifies the last address space entered.

Examples

To list all members named TIME in the SYSPROC concatenation for your address space, type:

```

findmbr time sysproc
      ①          ②          ③
AMTFMII JOBNAME: TS01          DDNAME: SYSPROC          MEMBER: TIME
AMTFMVI =====
      ④          ⑤          ⑥          ⑦          ⑧
AMTFM2I CT          LAST UPDT          USERID          DATA SET
AMTFMLI 1 12SEP91 14:17 BKE1          SYSP. STD. I SPPLIB
AMTFMLI 3 21AUG93 08:57 IMB1          SYS2. BOOL. BBCLIB
  
```

Legend:

1. Jobname.
2. DD name.

3. Member name.
4. Concatenation number of library within the specified DD name.
5. Date member was last updated.
6. Time member was last updated.
7. ID of user who updated the member last.
8. Name of data set.

To list all RESXA Proc names available through the alternate JES2, type:

```

findmbr resxa proclib jes2a

AMTFMII JOBNAME: JES2A      DDNAME: PROC00  MEMBER: RESXA
AMTFMVI =====
AMTFM2I CT      LAST UPDT  USERID  DATASET
AMTFMLI 1      02JAN94 14:17 REK1     SYS1. PROCLIB
AMTFMLI 4      24MAR94 08:57 SLG1     SYS2. BOOL. PROCLIB

AMTFMII JOBNAME: JES2A      DDNAME: PROC01  MEMBER: RESXA
AMTFMVI =====
AMTFM2I CT      LAST UPDT  USERID  DATASET
AMTFMLI 1      02FEB91 14:17 DEW1     SYSB. PROCLIB
AMTFMLI 4      01APR91 08:57 LAW1     SYS2. BOOL. SYSB. PROCLIB

AMTFMII JOBNAME: JES2A      DDNAME: PROC02  MEMBER: RESXA
AMTFMVI =====
AMTFM9I MEMBER RESXA NOT FOUND IN PROC02
    
```

To list the data sets that are allocated with DD name LOADLIB to the address space DWPBPAS and contain load module LGS, type:

```

findmbr lgs loadlib dwbpas

AMTFMII JOBNAME: DWPBPAS    DDNAME: LOADLIB  MEMBER: LGS
AMTFMVI =====
AMTFM3I CT  ①  SIZE  ②  ALIAS OF  ③  AC  DATASET
AMTFMLI 1  30360  MAIN  255  SYS1. I SPPLIB
AMTFMLI 3  30148  MAIN  255  SYS2. BOOL. BBPLIB
    
```

Legend:

1. Number of bytes in load module.
2. Alias name of member.
3. Authorization code for the module.

INFO

The INFO (IN) service displays information about the current operating environment. This service lets you list the release level of the operating system, the serial number and model of each CPU, and the type of IPL last performed.

Syntax

```
▶▶——I N F O——▶▶
```

Example

To display information about the current operating environment, type:

```

info
RESOLVE PLUS Release 3.1.1
MVS RELEASE SP6.0.6  FMID HBB6606  Mode GOAL      ❶
CPU 0 Serial Number 023481 Model 9672           ❷
CPU 1 Serial Number 123481 Model 9672
CPU 2 Serial Number 223481 Model 9672
CPU 3 Serial Number 323481 Model 9672
CPU 4 Serial Number 423481 Model 9672
CPU 5 Serial Number 523481 Model 9672
LAST IPL was COLD Start (CLPA) on 02/05/1999 at 1:21:42 from 026G11 ❸
IPL used LOADC6 in SYS1.IPLPARM on volume TSG316 (7D00)
IEASYM list = C6
IEASYS list = (C6), Source=Oper
TSO Version 2 Release 6 Mod 0                    ❹
VTAM Terminal ID B038T330                        ❺
Common Storage Tracking: CSA=ON SQA=ON

Volser PARMLIB Data Set Name
-----
SYS1.PARMLIB

Current LNKLST set is OS390.LNKLSTF1

```

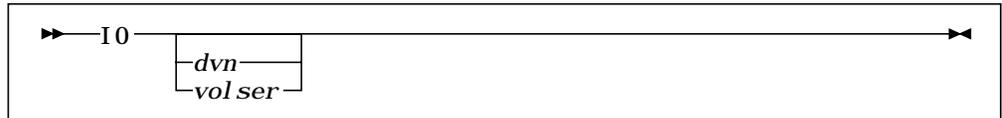
Legend:

1. OS/390 (MVS) release and its FMID.
2. Serial number for each CPU and the CPU model.
3. Type of IPL most recently performed (cold start, warm start, quick start).
4. Current release of TSO/E.
5. VTAM (or TCAM) logical terminal ID.

IO

The IO service detects potential device performance delays by displaying all outstanding non-TP I/O. Use this service to determine whether an address space is hung up because of an incomplete I/O operation, or to determine whether a device is in error recovery.

Syntax



where

dvn Is the device number for which outstanding non-TP I/O is to be displayed.

vol ser Is the volume serial number for which outstanding non-TP I/O is to be displayed.

Note: If you do not specify *dvn*, the IO service monitors all outstanding non-TP I/O.

Examples

To display all outstanding non-TP I/O, type:

```

io
  1 2 3
AMTI 11I INVENTORY UNIT 223 PACK08 IOQ 00F832A0 DRIVER- EXCP CCHH- 191 3
  4 5 6 7
AMTI 11I * TEST UNIT 180 333214 IOQ 00F85A00 DRIVER- VSAM
AMTI 11I + *MASTER* UNIT 1A4 PAGEL1 IOQ 00F91300 DRIVER- ASM CCHH- 271 12
    
```

Legend:

1. A one-character flag. Valid characters are as follows:
 - blank Line represents the I/O request currently active on the device.
 - * The I/O request on this line has been queued by the I/O supervisor pending completion of a previous I/O request.
 - + An ASM channel program is active on this device.
2. Address space name.
3. Device number.
4. Volume serial of the device, if it is a direct access storage or tape device.
5. IOQ address.

6. Name of IOS driver. Valid driver codes are:

Code	Driver
MISC	Miscellaneous driver
EXCP	EXCP driver
VSAM	VSAM driver
VTAM	VTAM driver
TCAM	TCAM driver
OLTEP	OLTEP driver
PCIFETCH	Program FETCH driver
JES3	JES3 subsystem
IOSPURGE	Internal IOS PURGE IOQ routine
VPSS	Vector processing subsystem (3838 array processor)
CRYPTO	Cryptographic subsystem
ASM	Auxiliary storage manager (paging supervisor)
DYNPATH	Path reconfiguration
SVC-33	IO HALT I/O SVC routine
R-CLEAR	Clear device recovery
R-SUBCHN	Subchannel recovery
SVCPURGE	IO PURGE I/O SVC routine
ALTPATH	Alternate path recovery
MIH	Missing interrupt handler
UNKNOWN	Unknown or unassigned driver code in use

7. Seek address in CCHH format (DASD device only).

To display IO for volume PACK08, type:

```

io pack08
AMTI 11I   GOJOB UNIT 283 PACK08 IOQ 00F83DA0 DRIVER- EXCP CCHH- 191 3①
                ②                ③                ④
AMTI 12I   UNIT B00: RESERVES- 0  ALLOCATIONS- 3  OPENS- 2
  
```

Legend:

1. Device number, volume serial, IOQ and IOSB addresses of the I/O, the OS/390 I/O driver, and the seek address for this I/O operation.
2. Number of outstanding RESERVE requests for this device.
3. Number of address spaces allocating this device.
4. Number of open DCBs against this device.

Note: Message AMTI12I is displayed for DASD devices only.

IPLDATA

The IPLDATA (IPL) service displays information obtained from the Initialization Parameter Area (IPA), which is mapped by IBM's IHAIPA macro.

Note: The IPL time, obtained from the IPA, may be several minutes after the start of system initialization.

Syntax



For IEASYS parameters, the Source column indicates the source of the parameter and can be:

- The name of the parmlib member from which it was obtained
- OPER , indicating that it was specified by the operator
- DEFAULT, indicating that the system default was used because the parameter was not specified

Example

To execute IPLDATA for IEASYS parameters, type:

```

ipl
LPAR=SYSC, IPL on Wednesday, December 16, 1998 at 0:50:30 AM
Loadparm Unit=7D00, Volser=TSG316, SYS1.IPLPARM(LOADC6)
IODF      04 SYSP      IODF00C1 C1
SYSCAT    TSG318113CI CFMCAT. SYSC. 0S390260
IEASYM    C6
SYSPLEX   BBPLEX01 X
NUCLEUS   1
NUCLST    00 N

Parmlib   Unit= 2B2, Volser=TSG321, SYS1.PARMLIB

IEASYS=

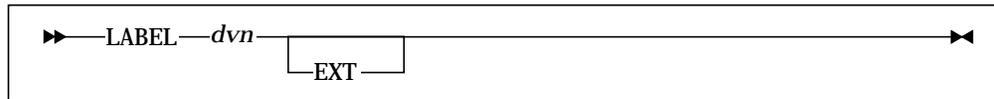
      Source Parm                IEASYSxx Parameter Value
-----
Default ALLOC
Default APF
IEASYSC6 APG                    07
Default BLDL
Default BLDLF
IEASYSC6 CLOCK                  00
Default CLPA
IEASYSC6 CMB                    (UNITR, COMM, GRAPH, CHRDR)
    
```

IEASYSC6	CMD	C3
IEASYSC6	CON	
Default	CONT	
IEASYSC6	COUPLE	SC
Default	CPQE	
IEASYSC6	CSA	(3548, 300M)
IEASYSC6	CSCBLOC	ABOVE
Default	CVIO	
Default	DEVSUP	
IEASYSC6	DIAG	SC
IEASYSC6	DUMP	DASD
Default	DUPLEX	
Default	EXIT	
IEASYSC6	FIX	C6
IEASYSC6	GRS	STAR
Default	GRSCNF	00
IEASYSC6	GRSRNL	00
IEASYSC6	ICS	PT
IEASYSC6	IOS	00
Default	IPS	00
Default	LNK	00
IEASYSC6	LNKAUTH	APFTAB
IEASYSC6	LOGCLS	L
IEASYSC6	LOGLMT	999999
Default	LOGREC	SYS1. LOGREC
IEASYSC6	LPA	(26, C6)
IEASYSC6	MAXCAD	60
IEASYSC6	MAXUSER	370
IEASYSC6	MLPA	C4
IEASYSC6	MSTRJCL	00
Default	NONVIO	
IEASYSC6	NSYSLX	100
Default	NUCMAP	
IEASYSC6	OMVS	C6
IEASYSC6	OPI	YES
IEASYSC6	OPT	PT
Default	PAGE-OP	
IEASYSC6	PAGE-SYS	(PAGE. VPAGC31. PLPA, PAGE. VPAGC31. COMMON, PAGE. VPAGC31. LOCAL1)
Default	PAGNUM	
IEASYSC6	PAGTOTL	(20, 20)
IEASYSC6	PAK	00
IEASYSC6	PLEXCFG	MULTISYSTEM
IEASYSC6	PRODP	C5
IEASYSC6	PROG	(26, C6, 0L)
Default	PURGE	
Default	RDE	NO
IEASYSC6	REAL	128
Default	RER	NO
IEASYSC6	RSU	0
IEASYSC6	RSVNONR	25
IEASYSC6	RSVSTRT	25
IEASYSC6	SCH	00
IEASYSC6	SMF	00
IEASYSC6	SMS	SC
IEASYSC6	SQA	(6, 40)
IEASYSC6	SSN	SC

LABEL

The LABEL (LA) service reads a tape label and displays its contents. This service lets you find the data set name and other characteristics specified in the tape label.

Syntax



where

dvn Is the hexadecimal device number of the tape drive.

EXT Causes a full 80 bytes of label information to be displayed.

Note: An INTERVENTION REQUIRED message appears on the console for the tape drive on which the tape is to be mounted.

Example

To display the contents of the tape label on device 180, type:

```

Label 180
AMTL12I VOL1000111 OWNRI D ①
AMTL12I HDR1USER. DATASET 00000100010001 091775 ③
AMTL12I HDR2F035200008031MYJOB /STEPONE ④
    
```

Legend:

1. Volume label.
2. Data set name.
3. Header label #1.
4. Header label #2.

LCPU

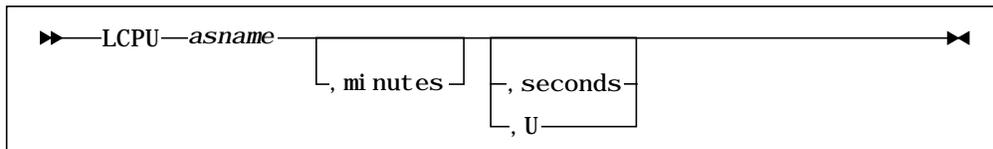
The LCPU service provides the ability to display and modify the CPU time limits for an address space that were originally specified by the TIME parameter on the JOB and EXEC statements.

The CPU limit for the current job step and the limit for the job (maximum for all job steps) are displayed. The CPU time used and CPU time remaining for the job step are also displayed.

If new values are specified, the step limit is reset to the specified limit value. The job limit is also reset if the current value is less than the specified value; otherwise it remains unchanged.

The maximum CPU limit value that may be specified is 1,440 minutes, which may be specified in minutes and/or seconds.

Syntax



where

asename Is the address space name. You may specify a jobname or an ASID. To specify an ASID, type (*nn*) where *nn* is the ASID in decimal or type (*Xnn*) where *nn* is the ASID in hexadecimal. The default is the address space of the issuer.

(, minutes) Is the number of minutes for the new CPU time limit. The maximum value that may be specified is 1440.

(, seconds) Is the number of seconds for the new CPU time limit. You may specify the CPU time limit in minutes and/or seconds. If specifying in seconds only, you must insert two commas between the *asename* and the number of seconds to indicate zero minutes.

(, U) A U in the third or fourth parameter position indicates that the CPU time limit for the step may be set to a value less than the current limit value.

Examples

To display the current CPU time limits and the amount of time remaining for the job and job step, type:

```

1 lcpu, vamb3a
AMFLC1I CPU limit for job VAM3A 4 3 10 mi nutes
AMFLC2I CPU limit for current step ASMSTEP 4 2 mi nutes, 30 seconds
AMFLC3I Step time used 4 1 mi nutes, 35 seconds
AMFLC4I Step time remain ing 4 55 seconds
AMFLC0I CPU limit changed 5
    
```

To increase the CPU time limit for the current job step and to display the new limit values and remaining CPU time, type:

```

1 lcpu, vamb3a, 5
AMFLC1I CPU limit for job VAM3A 4 3 10 mi nutes
AMFLC2I CPU limit for current step ASMSTEP 4 5 mi nutes
AMFLC3I Step time used 4 1 mi nutes, 35 seconds
AMFLC4I Step time remain ing 4 3 mi nutes, 25 seconds
AMFLC0I CPU limit changed 5
    
```

Legend

1. Message number.
2. Address space name (asname).
3. Minutes and seconds fields.
4. CPU time limit message lines.
5. Notification of changed CPU limit.

LLIST

The LLIST (LL) service

- Lists the names of all libraries in the link list and the number of extents that exist for each library when the link list is built
- Determines which link list library or libraries contain a specified module
- Lets you dynamically rebuild the link list

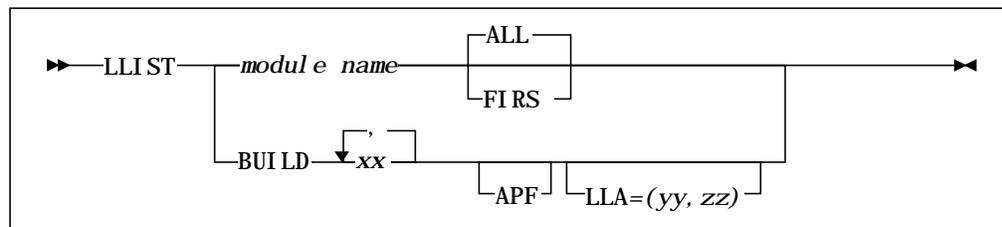
Note: The LLIST service provides support for systems prior to OS/390 1.3. The LNKLST service provides support for the dynamic LNKLST facility implemented in OS/390 1.3.

The operating system does not let you add or delete link list data sets without IPLing the system. New modules added to a link list data set may exceed the free space within the existing extents, causing a new extent to be added to the data set. Abend 106-F results when these new modules are accessed because they reside outside of the extents that existed when the link list was created. Normally, an IPL is required to permit access to these new modules. The LLIST service lets you to rebuild the link list dynamically, eliminating the need for an IPL.

If LLA manages the link list, you must use the LLA parameter when you build the link list. The LLA parameter includes two suffixes, *yy* and *zz*. These suffixes refer to CSVLLA members that reside in the library pointed to by IEFPARM in the LLA address space (by default, SYS1.PARMLIB). CSVLLA_{yy} removes the link list from control of the LLA address space before the BUILD process; CSVLLA_{zz} places the link list back under control of the LLA address space after the BUILD process is completed.

See the Usage Notes for more information on the LLA parameter.

Syntax



where

<i>module name</i>	Is the name of the module to search for in the link list libraries.
ALL	Causes all libraries containing the module to be listed; the default.
FIRST	Stops the search at the first occurrence of <i>module name</i> .
BUILD	Creates a new dynamic link list.
xx	Is a two-character suffix of a LNKLST member in SYS1.PARMLIB used to create the new link list. You can specify from one to seven suffixes. There is no default.

- APF** Is an optional parameter that signals you to use the in-storage APF list to determine whether data sets should be authorized. If you omit this parameter, all link list data sets are authorized.
- yy** Is the suffix of a CSVLLA member that specifies the removal of the link list data set concatenation from control of the LLA address space before a new dynamic link list is created.
- zz** Is the suffix of a CSVLLA member that places the link list data set concatenation back under control of the LLA address space after a new dynamic link list is created.

Note: Use suffixes yy and zz only if your system uses LLA. These suffixes refer to CSVLLA members that reside in the library pointed to by IEFPARM in the LLA address space. By default, this library is SYS1.PARMLIB.

Examples

To list all data set names in the link list, type:

```

llist
AMTL76I LINKLIST HAS BEEN REBUILT ON 01/23/96 AT 12:34:55 BY TS0ID1
      ①      ②      ③      ④
AMTL70I VOLSER EXT APF DATASET NAME
AMTL71I SYSR1C 1 YES SYS1. LINKLIB
AMTL71I SYSR1C 1 YES SYS1. MGLIB
AMTL71I SYSR1C 1 YES SYS1. CMDLIB
AMTL71I BAB012 1 NO SYS1. CORP. BBLINK
AMTL71I BAB031 1 NO SYS1. CORP. BBLOAD
AMTL71I TSG006 1?YES SYS2. USER. SHRDLIB
AMTL72I ** WARNING ** DATASET NOW HAS 4 EXTENT(S) ⑤
AMTL71I TSG006 1 YES SYS2. SYSB. SHRDLIB
AMTL71I TSG005 1 YES SYS2. SHRD. SHRDLIB
AMTL71I BAB014? 2 NO NML1. ASM LOAD
AMTL77I ** WARNING ** DATASET IS NOW CATALOGED ON BAB048 ⑥
AMTL71I SYSR2C 1 NO PLI. PLILINK
AMTL71I TSG006 1 NO SYS2. V2R1M0. GDDMLoad
    
```

Legend:

1. Volume where link list library resides at the time the link list is built.
2. Number of extents allocated at the time the link list is built.
3. Indicates whether data set is APF-authorized.
4. Name of the data set.
5. Issued if additional extents are added to this data set after the link list is built.
6. Issued when the data set is recataloged to a different volume, which occurs after the link list is built.

To list all link list data set names that contain module IEFACTRT, type:

```

llist iefactrt

AMTL73I MODULE IEFACTRT FOUND IN STEPLIB DATASET
AMTL74I MODULE IEFACTRT FOUND IN LINKLIST DATASET(S)
AMTL70I VOLSER EXT APF DATASET NAME
AMTL71I SYSR1C 1 YES SYS2. CORP. BBLINK
AMTL71I SYSR1C 1 NO SYS2. SHRD. SHRDLIB

```

Legend:

1. Issued if the specified module is also found in the STEPLIB concatenation.

To build the link list using the definitions contained in SYS1.PARMLIB members LNKLST00, LNKLST01, and LNKLST33, using the in-storage APF list to determine whether data sets should be authorized, type:

```

llist build 00 01 33 apf
AMTL80I VOLSER EXT APF MBR DATASET NAME
AMTL81I SYSR1C 1 YES SYS1. LINKLIB
AMTL81I SYSR1C 1 YES SYS1. MGLIB
AMTL81I SYSR1C 1 YES 00 SYS1. CMDLIB
AMTL81I BAB012 1 NO 00 SYS1. CORP. BBLINK
AMTL81I BAB031 1 NO 00 SYS1. CORP. BBLOAD
AMTL81I TSG006 2 YES 01 SYS2. SYSB. SHRDLIB
AMTL81I TSG005 1 YES 01 SYS2. SHRD. SHRDLIB
AMTL81I TSG006 13 YES 01 SYS2. USER. SHRDLIB
AMTL81I TSG006 1 NO 33 SYS2. V2R1MO. GDDMLoad
AMTL81I SYSR2C 1 YES 33 PLI. PLILINK
AMTL81I BAB023 1 NO 33 NML1. TEST. LINKLIB
AMTL8KA REPLY Y TO CONFIRM NEW LINKLIST, N TO CANCEL INPUT
y
AMTL8CO PLEASE WAIT A MOMENT WHILE LLA PROCESSES LNKLST
AMTL8EI NEW LINKLIST DATASETS ARE NOW ACTIVE

```

Legend:

1. Suffix for the LNKLST member of SYS1.PARMLIB that contains the definition for each data set.
2. Name of the data set.

Note: To activate the new link list, type Y in response to message AMTL8KA.

Usage Notes

- SYS1.LINKLIB must be the first data set in the link list; it is added automatically as the first data set and is ignored in any other location. The new link list can contain a maximum of 255 extents.
- If you are using a third-party software program with a function similar to LLA, stop that program before creating a new dynamic link list and start it again after creating a new dynamic link list.

- If you are using LLA on your system and do not specify the *yy* and *zz* suffixes, LLIST uses the default suffixes in BBPARM library member \$\$INSYS0. You have the option to define these default suffixes during the AutoCustomization process.
- If you have not defined the *yy* and *zz* suffixes in BBPARM library member \$\$INSYS0, and omit them when you type the LLIST command, the LLIST service will build the link list only:
 - If you are not using LLA
 - In an ESA environment where the link list is not under control of LLA

LNKLST

The LNKLST service provides the ability to create, modify, and delete LNKLST sets; make a LNKLST set the active (current) set; switch address spaces to the current set; and display related information.

The LNKLST service will also search the user's JOBLIB/STEPLIB, the LPA, and the current LNKLST set for a specified load module and display where it was found.

Note: The LNKLST service provides support for the IBM dynamic LNKLST facility implemented in OS/390 1.3 and, therefore, is not supported on earlier operating systems. However, the LLIST service continues to provide similar functions for systems prior to OS/390 1.3.

General Information

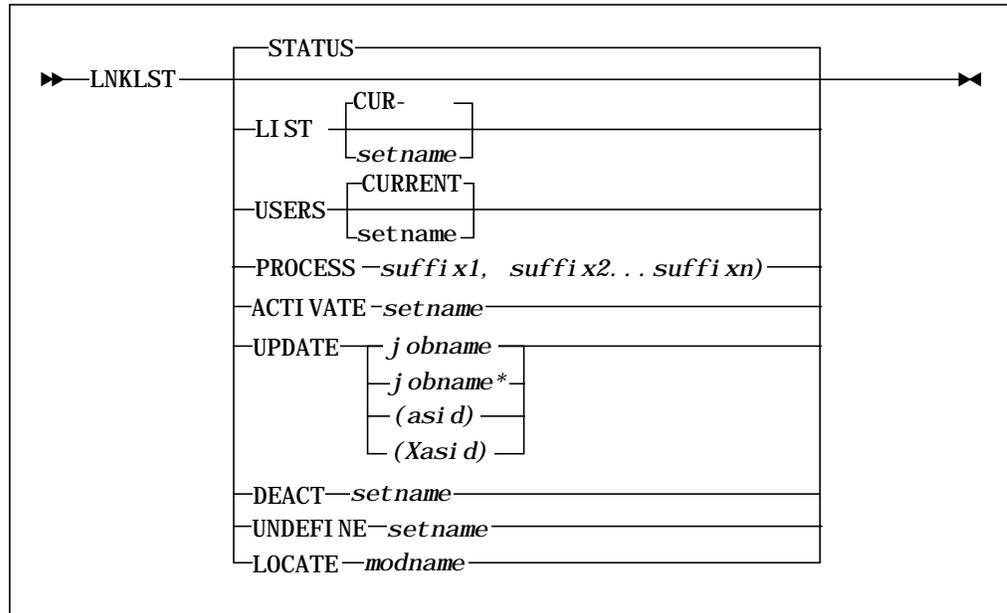
The IBM Dynamic LNKLST facility introduced the concept of LNKLST sets. Each address space in the system is assigned to a LNKLST set. A LNKLST set defines the data sets that compose the LNKLST for the address spaces using that LNKLST set. LNKLST set names are 1 to 16 characters in length. The initial set is named IPL. The IPL set cannot be modified. New address spaces are assigned to the CURRENT set. The name CURRENT is generic, and refers to the current LNKLST set, regardless of its actual name.

You cannot modify or UNDEFINE a LNKLST set that has address spaces assigned to it. You may use the LNKLST DEACT function or the LNKLST UPDATE function to switch address spaces to the current LNKLST set. You can use the LNKLST ACTIVATE function to make the specified LNKLST set the current LNKLST set. You can use the LNKLST PROCESS function to copy an existing LNKLST set, add data sets to an existing set, or delete data sets from an existing set. You can use the LNKLST UNDEFINE function to remove an existing LNKLST set from the system.

The functions STATUS and USERS display information about the existing LNKLST sets and the address spaces assigned to them.

The LOCATE function searches the data sets in the current LNKLST set for the specified load module and displays the name of the data set containing the load module.

Syntax



Note: All function keywords (first parameter) may be abbreviated to their first three letters.

where

STATUS	Displays a list of the defined LNKLST sets and the number of address spaces using each LNKLST set. A plus sign (+) preceding a set name indicates that it is the current LNKLST set. A minus sign (-) preceding a set name indicates that the set had been the current set at one time.
LIST	Displays a list of the data sets in the specified LNKLST set and their sequence number, volser, APF, and SMF status. If a set name is not specified, the current set is displayed.
USERS	Displays a list of address spaces currently assigned to the specified LNKLST set. If a set name is not specified, the current set is displayed.
PROCESS	Allows you to process a LNKLST set. You must specify at least one suffix.
ACTIVATE	Allows you to make the specified LNKLST the current set. You must specify a LNKLST.
UPDATE	Switches one or more address spaces to the current LNKLST set. You must specify a jobname or ASID.
DEACT	Switches all address spaces assigned to the specified LNKLST set to the current LNKLST set.
UNDEFINE	Removes the specified LNKLST set from the system.
LOCATE	Searches the JOBLIB/STEPLIB, LPA, and the current LNKLST set for the specified load module and displays its location.

Examples

This section provides examples on how to use the LNKLST functions.

LIST

To display a list of the data sets in the IPL set, type:

```
LNK, LIST, IPL
```

To display a list of the data sets in the current LNKLST set, type:

```
LNK, LIST  
or  
LNK, LIST, CURRENT
```

USERS

The USERS function displays a list of the address spaces currently assigned to the specified LNKLST set. If a set name is not specified, information for the current LNKLST set is displayed. The information includes the set name, current number of address spaces, the ASID (in hexadecimal), and the jobname for each address space.

To display a list of the address spaces using the IPL set, type the following command:

```
LNK, USE, IPL
```

To display a list of the address spaces using the current set, type the following command:

```
LNK, USE
```

PROCESS

The primary purpose of PROCESS is to create new LNKLST sets and modify existing sets. However, the functions UPDATE and UNDEFINE are supported as well.

PROCESS reads the specified PROGxx members and processes the following statements:

```
LNKLST DEFINE  
LNKLST ADD  
LNKLST DELETE  
LNKLST UPDATE  
LNKLST UNDEFINE
```

Other statements (APF, EXIT SYSLIB, LNKLST TEST, and LNKLST ACTIVATE) are ignored.

Specify the suffixes of the PROGxx members that you want to process as the second parameter of the command. If you specify more than one suffix, you must enclose them in parentheses. For example, the following command will process members PROG01 and PROG03:

```
LNKLST, PRO, (01, 03)
```

Creating a new LNKLST set or modifying an existing set is a two-step process. First, you need to create a PROGxx member in a data set included in the logical PARMLIB concatenation (formerly SYS1.PARMLIB). Second, use LNKLST, PROCESS to process the statements in PROGxx.

To identify the data sets that are to be included in the new or modified LNKLST set, use the PROGxx statements:

```
LNKLST DEFINE  
LNKLST ADD  
LNKLST DELETE
```

Note: If your PROGxx member contains symbols, they are replaced with their current value.

You want to add a new data set called SYS2.MYLIB to the beginning of the current LNKLST, which is the IPL LNKLST set. Since you cannot modify the IPL set, you create your own LNKLST set (called MYSET) by copying the IPL set, and then adding your data set to the copy.

First, create PARMLIB member PROGMY containing a LNKLST DEFINE statement to define the new set and copy the data sets from the IPL set. Also include a LNKLST ADD statement to add your data set after SYS1.CSSLIB.

Note: See the IBM publication, *OS/390 MVS Initialization and Tuning Reference*, for an explanation of the LNKLST statement used in member PROGxx.

```
PROGMY  
  
lnklst define name(myset) copyfrom(ipl)  
lnklst add name(myset) dsname(sys1.mylib) attop
```

Then use the following LNKLST command to process the member you have created:

```
LNK, PRO, MY
```

At this point, you have created LNKLST set MYSET. It is available now, but will not be used until it has been activated. Once activated, new address spaces will be assigned to it, but the existing address spaces will continue to use the IPL set until you use either DEACT or UPDATE to switch address spaces to the current set.

If you want to have all address spaces use your new LNKLST set, you could execute the following commands to make LNKLST set MYSET the current set and transfer all address spaces using set IPL to the current set:

```
LNK, ACT, MSET
LNK, DEACT, IPL
```

ACTIVATE

The ACTIVATE function makes the specified LNKLST set the current LNKLST set.

Type the following command to make LNKLST set MYSET the current LNKLST set:

```
LNK, ACT, MSET
```

UPDATE

The UPDATE function switches one or more address spaces to the current LNKLST set. You may specify a jobname or an ASID. If you specify a jobname, you may use the wild card characters * and ? to select a group of jobs. An ASID may be specified in hexadecimal or decimal format. Indicate that the value is an ASID by enclosing it in parentheses. Indicate that it is a hexadecimal value by prefixing it with an X. Therefore, (123) is a decimal ASID and (X123) is a hexadecimal ID, equivalent to (291).

Type the following command to switch all address spaces to the current LNKLST set:

```
LNK, UPDATE, *
```

Type the following command to switch all jobs (and TSO users) starting with jobnames beginning with the letters ABC to the current set:

```
LNK, UPD, ABC*
```

Type the following command to switch all jobs (and TSO users) with ABC as the first three characters of the jobname, any character in the fourth position, a 1 in the fifth position, and any characters in the remaining three positions:

```
LNK, UPD, ABC?1*
```

DEACT

The DEACT function switches all address spaces assigned to the specified LNKLST set to the current LNKLST set.

Type the following command to switch all address spaces assigned to LNKLST set MYSET to the current set:

```
LNK, DEACT, MSET
```

UNDEFINE

The UNDEFINE function removes the specified LNKLST set from the system. A LNKLST set cannot be removed if any address spaces are using it. In addition, the IPL set cannot be removed.

Note: You can use DEACT to switch all of the address spaces to the current set.

Type the following command to remove LNKLST set MYSET from the system:

LNK, UND, MYSET

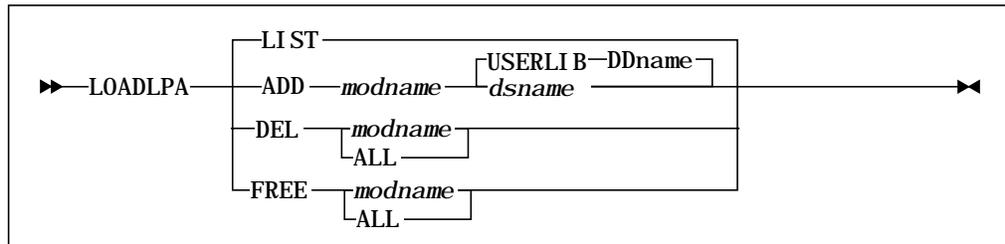
LOCATE

The LOCATE function searches the JOBLIB/STEPLIB, LPA, and the current LNKLST set for the specified load module. If the module is found in the JOBLIB/STEPLIB concatenation, the concatenation number is displayed (the first data set is number 1). If the load module is found in the current LNKLST set, the name of the data set in which it was found is displayed.

LOADLPA

The LOADLPA (LOA) service lets you add modules or SVCs in LPA without needing to re-IPL. This service also lets you delete modules added by this service.

Syntax



where

- LIST** Lists information for all active modules. A module is considered active if it was loaded by the ADD function and has not been deleted. The LIST function is the default.
- ADD** Loads a module and all of its aliases; it also places the module on the active LPA queue. Modules are loaded from the load library with a DD name of USERLIB, unless a data set name is specified as the third parameter.
- If you specify a module that was previously loaded by the ADD function, the module is loaded at a new location because the entry point changes. The previously loaded copy is deleted (see the description of the DEL function).
- modname* Is the name of the module that is to be added, deleted, or freed.
- dsname* Is the name of the data set where *modname* resides. LOADLPA dynamically allocates the data set specified before loading *modname* into LPA. If no data set name is specified, LOADLPA tries to find *modname* in the DD concatenation USERLIB. Note that USERLIB must have been previously allocated to the same address space where SYSPROG Services is currently executing. (When SYSPROG Services is running under MainView for OS/390 from the SYSPROG Services menu, this is the PAS address space.)
- DEL** Deletes a module and its aliases loaded by the ADD function. All saved entry point addresses are restored and the module is removed from the active LPA queue. The module's storage is freed if the module's use count is zero and if the module does not contain an SVC entry point. See the description of the FREE function for information about how to recover storage for these modules.
- If you specify ALL, LOADLPA deletes all modules and aliases previously loaded by the ADD function.

FREE Frees storage for a deleted module whose use count is zero. Storage is not freed when the use count is greater than zero since a non-zero use count implies the module is still in use.

You may repeat the LOADLPA FREE command at a later time. You can also use this function when the deleted module contains an entry point for an SVC.

If you specify ALL, then deleted modules with use counts of zero are freed.

Examples

To add the USERMOD module, which is found in LGS1.LOADLIB, type:

```

loadlpa add usermod lgs1.loadlib
AMTL5HI 1NAME=USERMOD, 2EP=00B41000, 3CDE=F47AD0, 4TIME=17:13
AMTL5JI 5LA=00BA4100, 6LEN=0022CE
AMTL5KI ALIAS=USEREP1, EP=00BA4140, CDE=F47B00 7
AMTL5KI ALIAS=USEREP2, EP=00BA4440, CDE=F47B30 7
AMTL5LI LOADLPA ADD FUNCTION COMPLETED

```

Legend:

1. Name of module loaded.
2. Address of module entry point.
3. Address of the contents directory entry (CDE) created for this module.
4. Time module was loaded.
5. Address of module.
6. Length of module.
7. Name, entry point address, and CDE address for all aliases found in the specified module for this module.

To delete the USERMOD module from LPA, type:

```

loadlpa del usermod
AMTL65I MODULE USERMOD DELETED
AMTL5PI LOADLPA DELETE FUNCTION COMPLETED

```

To free module storage if the use count in the major CDE is reduced to zero, or to free storage for a module that contains an SVC entry, type:

```

loadlpa free usersvc
AMTL67I MODULE USERSVC FREED
AMTL6AI LOADLPA FREE FUNCTION COMPLETED

```

Usage Notes

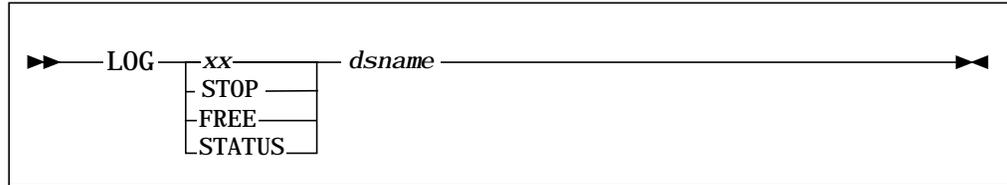
- Entry points of some operating system routines are maintained in system control blocks. The entry points used for specific modules and control blocks depend on the operating system, release level, and maintenance level. The LPA active queue is the only control block that LOADLPA updates. Therefore, programs that do not use LINK, LOAD, ATTACH, or XCTL to access the module are unaffected by LOADLPA.
- LOADLPA can delete only the module and aliases that LOADLPA loads.
- The SVC table is updated with an entry point address for any module name (true name or alias) that conforms to SVC naming conventions. If necessary, you can use the ZAP service to update the SVC table to change the locks. All types of extended SVCs are supported. In all cases, the old entry point address is saved and then restored when the module is deleted by the LOADLPA delete function. Use the ZAP service to reset the locks. See [“SVCFIND” on page 218](#) and [“ZAP” on page 251](#) for more information.
- Using the LOADLPA service, you can either add to or delete from LPA the SVC names that end with X' C0' (such as type 4 SVC 250). To do this, substitute a left brace ({} for the last characters of the module name; for example:

LOADLPA ADD IGC0025{
- When you add an extended SVC with the LOADLPA service, the entry point address in the Extended SVC Router (ESR) table is updated. However, the flags in the second word of the ESR table entry are not changed. Use the ZAP service to reset flags, if required.
- Modules loaded by the LOADLPA service are logically added to the LPA, but physically reside in CSA storage.

LOG

The LOG service controls the logging system. The logging system provides the ability to periodically execute a set of services with the output directed to a data set that you specified.

Syntax



where

- xx* Is the suffix of the \$\$INLGxx member used to start logging. The \$\$INLGxx member contains control statements that define the services to be executed and control the logging process. This parameter also implies start.

- dsname* Is the name of the partitioned data set that contains the \$\$INLGxx member. If this parameter is omitted, the name you last specified is used. If you did not previously specify a *dsname*, the data set allocated to DDNAME BBPARM is used.

- STOP Indicates that logging is to be stopped.

- FREE Indicates that the logging output data set is to be freed. FREE implies stop.

- STATUS Indicates that the status of the logging system is to be displayed.

\$\$INLGxx Control Statements

The verb (statement type) must begin in column 1. One or more blanks are allowed between the verb and the operand. Operands cannot extend beyond column 72.

- CLASS *c* Indicates that the logging output is to be written to a SYSOUT data set of the specified class. CLASS is mutually exclusive with the DSN statement.

- CMD *Command* Is a command that is to be periodically executed. Use multiple CMD statements for multiple commands. Commands are executed in the order that they appear.

- CNT *n* Specifies that logging is to be terminated when *n* number of cycles have been executed. Also, see the STOP statement.

- DSN *dsname* Specifies that logging output is to be written to the preallocated data set *dsname*. If no CLASS or DSN statement is provided, and a data set has been allocated to DDNAME LOG, the output will be written to that data set. Otherwise, it will be written to SYSOUT=A.

EJC	Indicates the output produced by the command that follows and it should start on a new page. The EJC statement should be followed by one or more CMD statements.
INCR <i>n</i>	Indicates the number (<i>n</i>) of seconds (increment) in an interval. The default is 30 seconds. The product of INCR and INT determines the number of seconds between logging cycles.
INT <i>n</i>	Indicates the number (<i>n</i>) of intervals in a logging cycle. The default interval is one. The product of INCR and INT determines the number of seconds between logging cycles.
START <i>hhmm</i>	Indicates the time (<i>hhmm</i>) that logging is to start. Specify 1300 for 1:00pm. Logging starts immediately if you do not provide a START statement.
STOP <i>hhmm</i>	Indicates the time (<i>hhmm</i>) that logging is to terminate. Specify 1200 for 12:00pm (noon) and 2400 for 12:00am (midnight). Note: If you provide both CNT and STOP statements, logging will terminate when the first condition is reached.
HOLD	Indicates that the SYSOUT output is to be held. If HOLD and NOHOLD are not specified, the default for the SYSOUT class will prevail. The HOLD statement is ignored if logging output is not written to a SYSOUT data set.
NOHOLD	Indicates that the SYSOUT output is <i>not</i> to be held. If HOLD and NOHOLD are not specified, the default for the SYSOUT class will prevail. The NOHOLD statement is ignored if logging output is not written to a SYSOUT data set.
MAXOUT <i>n</i>	Indicates the maximum number (<i>n</i>) of lines that will be written to a SYSOUT data set. Logging will be terminated when <i>n</i> lines have been written to SYSOUT. The default is to write an infinite number of lines. MAXOUT is only applicable to SYSOUT data sets.

Examples

Member \$\$INLG01

```
INCR 60
INT 5
DSN VAM3.LOG
START 2200
STOP 2259
CMD CSA, MAP
```

Member \$\$INLG01 will cause the command CSA, MAP to be executed once every five minutes between 10:00pm and 10:59pm. The output will be written to data set VAM3.LOG.

Member \$\$INLG02

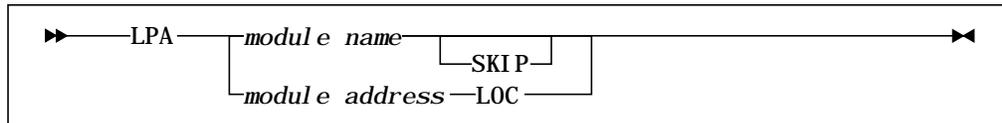
```
INCR 30
INT 2
CNT 60
HOLD
CLASS A
CMD MTP
CMD ENQ
CMD CPU 5
```

Member \$\$INLG02 causes the commands MTP, ENQ, and CPU 5 to be executed once a minute for a total of 60 times. The output will be written to SYSOUT Class A and placed on the HOLD queue.

LPA

The LPA service finds modules that are located in any of the system link-pack areas. This service also determines whether a given address lies within the base or extended system link-pack area (LPA), fixed link-pack area (FLPA), or modified link-pack area (MLPA).

Syntax



where

<i>module name</i>	Is the name of the module you want to locate within the link-pack area.
SKIP	Finds the second occurrence of the module.
<i>module address</i>	Is a virtual address returning the name of the module occupying that address.
LOC	Is required when specifying an address you want to process.

Examples

To locate module IEFJRASP in the system link-pack areas, type:

```

lpa iefj rasp
AMTL41I  MODULE=1EFJRASP  [ EMLPA ]
AMTL42I  LA=019EEA28  EP=019EEA28  (AMODE- 31)  LEN=000928
AMTL43I  CDE=00FDD020
  
```

Legend:

- Requested module name.
- Location in the link-pack area where the module resides.

PLPA	Pageable link-pack area
EPLPA	Extended pageable link-pack area
MLPA	Modified link-pack area
EMLPA	Extended modified link-pack area
FLPA	Fixed link-pack area
EFLPA	Extended fixed link-pack area
- Load address for the module.

4. Entry-point address for the module.
5. Addressing mode of the module entry point.
6. Length of the module.
7. Address of the Link Pack Directory Entry (LPDE) or the CDE address is displayed. A CDE exists only when the module is on the Active Link Pack Area Queue (ALPAQ).

To find the next copy of module IEFJRASP that resides in the PLPA, type:

Ipa iefjrasp skip

```
AMTL41I MODULE=IEFJRASP [ PLPA ]
AMTL42I LA=1013A000 EP=1013A000 (AMODE- 31) LEN=000928
```

To locate module IDCAM01, which has an alias entry point of DEFINE, type:

Ipa define

```
AMTL40I ENTRY=DEFINE ① EPL0C=00E4F83C (AMODE- 24)
AMTL41I MODULE=IDCAM01 [ PLPA ]
AMTL42I LA=00E4F000 EP=00E4F100 (AMODE- 24) LEN=01A430
```

Legend:

1. Alias entry point to the module listed in the AMTL41I message.

To locate the module with the address 19A032A, type:

Ipa 19a032a loc

```
AMTL44I LOC 019A032A IS ① +00332A FROM EPLPA ENTRY IEECB905 ③
AMTL42I LA=0199D000 EP=0199D000 (AMODE- 31) LEN=00F7F0
AMTL43I CDE=00FDE130
```

Legend:

1. Address supplied as input.
2. Hexadecimal offset of the supplied address from the LPA entry point in which the address resides.
3. Name of the closest LPA entry point to the supplied address.

Usage Note

The LPA service sets the location pointer (*) to the specified entry point for the module. Therefore, you can use the DUMP service to display the storage beginning at the entry point by typing the command DUMP, *.

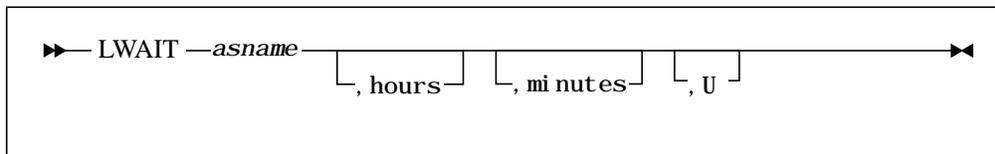
LWAIT

The LWAIT service displays the wait limit for an address space. If the address space is currently waiting, LWAIT also displays the amount of time it has been waiting since it was last dispatched, and the remaining wait time available before the address space will be terminated by the operating system with a 522 abend.

In addition, the LWAIT service provides the ability to increase or decrease the wait limit for the address space. To prevent errors, you must also specify U to decrease the wait limit.

Note that the wait limit is the amount of time the address space may remain in a continuous wait.

Syntax



where

<i>asname</i>	Is the address space name. You may specify a jobname or an ASID. To specify an ASID type (<i>nn</i>) where <i>nn</i> is the ASID in decimal or type (<i>Xnn</i>) where <i>nn</i> is the ASID in hexadecimal. This is a required field.
(, hours)	Number of hours for the new wait limit.
(, minutes)	Number of minutes for the new wait limit. You may specify the wait limit in hours and minutes or minutes only. If specifying in minutes only, you must insert two commas between the <i>asname</i> and the number of minutes to indicate zero hours.
(, U)	A U in the third or fourth parameter position indicates that the wait limit may be set to a value less than the current limit value.

Examples

To display the wait limit and the amount of time the job has been in a wait, type:

```

lwait, van3w1
  ①      ②
AMTLW1I VAM3W1
AMTLW4I Has been waiting for ⑥           ④      ⑤
AMTLW5I Wait limit remaining ⑥      1 hours, 4 minutes, 37 seconds
AMTLW6I Wait limit is        ⑥      2 hours, 55 minutes, 22 seconds
AMTLW6I Wait limit changed   ⑦      2 hours, 00 minutes
  
```

To increase the wait limit to three hours, type LWAIT, VAM3W1,3. The new wait limit is displayed.

```

1wait, van3w1, 3
  ①      ②      ③      ④      ⑤
AMTLW1I VAM3W1
AMTLW4I Has been waiting for ⑥      4 mi nutes, 58 seconds
AMTLW5I Wait limit remaining ⑥ 2 hours, 55 mi nutes, 22 seconds
AMTLW6I Wait limit is ⑥ 3 hours, 00 mi nutes
AMTLWoI Wait limit changed ⑦
    
```

The request to set the wait limit to 30 minutes was denied because the current wait limit is a larger value (one hour) and U was not specified.

```

1wait, van3w1, , 30
  ①      ②      ③      ④      ⑤
AMTLW1I Request denied. New value is less than old value.
AMTLW1I VAM3W1
AMTLW5I Has been waiting for ⑥      6 mi nutes, 45 seconds
AMTLW4I Wait limit remaining ⑥      53 mi nutes, 14 seconds
AMTLW6I Wait limit is ⑥ 1 hours, 00 mi nutes
AMTLWoI Wait limit changed ⑦
    
```

The wait limit is reduced to 30 minutes as requested. Note that U was specified to allow the reduction.

```

1wait, van3w1, , 30, u
  ①      ②      ③      ④      ⑤
AMTLW1I VAM3W1
AMTLW4I Has been waiting for ⑥      7 mi nutes, 10 seconds
AMTLW5I Wait limit remaining ⑥      22 mi nutes, 49 seconds
AMTLW6I Wait limit is ⑥      30 mi nutes
AMTLWoI Wait limit changed ⑦
    
```

Legend:

1. Message number.
2. Address space name (asname).
3. Hours field.
4. Minutes field.
5. Seconds field.
6. Wait limit message lines.
7. Notification of changed wait limit.

LX

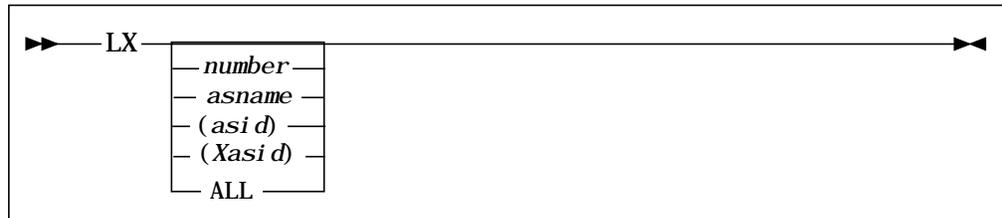
The LX service displays information for system linkage indexes (LXs). This service displays the maximum (total) number of system LXs, which are subdivided into dormant, available, and active.

The LX service displays the LXs assigned to the specified address space.

For dormant LXs, the LX number plus the jobname and ASID of the last address space to use the LX and the approximate time and date it became dormant are displayed when available.

Additionally, information can be displayed for all LXs, a specified LX, or all LXs assigned to a jobname or ASID.

Syntax



where

<i>lx number</i>	Is the LX number.
<i>asname</i>	Is the address space name.
<i>(asi d)</i>	Is the address space identifier in decimal format.
<i>(Xasi d)</i>	Is the address space identifier in hexadecimal format.
ALL	Indicates all system LXs.

Type	To
LX	Display the current system LX status and information for dormant LXs.
LX 23	Display information for LX number 23 (decimal).
LX ALL	Display information for all system LXs.
LX (X123)	Display information for all system LXs assigned to ASID x0123.
LX PCAUTH	Display information for all system LXs assigned to the specified address space.

Definitions

Total LXs	<p>The maximum number of system LXs.</p> <p>The total number of LXs is the sum of the number of dormant, available, and active LXs. The maximum number of system LXs is controlled by the parameter NSYSLX in PARMLIB member IEASYSxx.</p>
Dormant	<p>The address space that acquired or last used the LX has terminated.</p> <p>System LXs are not reassigned. However, the original requester can choose to reconnect to the LX at a future time.</p>
Available	<p>A system LX that has never been assigned.</p>
Active	<p>System LXs that have been assigned by the system and are not dormant.</p>
Owned	<p>An LX that has been assigned. It may either be active or dormant.</p>
LX	<p>Linkage Index number.</p>

MCOMMAND

The MCOMMAND (MCO) service executes master console commands from a TSO terminal and lets you control jobs remotely from a TSO terminal.

Syntax

```
▶▶ MCOMMAND — command ◀◀
```

where

command Is a command for OS/390 or JES.

Example

To execute the master console command DUMP COMM=(VTAM PROBLEM), type:

```
mcommand dump comm=(vtam problem)
```

```
AMTC21I    COMMAND    DUMP COMM=(VTAM PROBLEM) EXECUTED FROM CN (01)
```

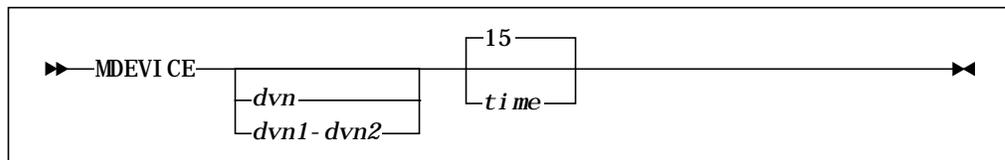
MDEVICE

The MDEVICE (MDEV) service monitors all or selected devices attached to all processors in the configuration sampled within a 15-second interval.

The MDEVICE service

- Monitors the I/O activity of all devices and address spaces (it can also monitor the I/O activity of a specified device or address space)
- Detects I/O bottlenecks that cause workload delays

Syntax



where

dvn Is a device number.

dvn1 - dvn2 Is a range of device numbers; *dvn1* is the lowest device number and *dvn2* is the highest device number.

time Specifies the length of the sample period in seconds. The default sample period is 15 seconds.

Examples

To monitor all devices for 15 seconds, type:

```

mdevice

AMT050I I/O DEVICE DATA BEING GATHERED

AMT051I I/O DEVICE ACTIVITY REPORT
      ①  ②  ③  ④  ⑤  ⑥  ⑦  ⑧  ⑨  ⑩
AMT036D -----
AMT036H DEV VOLSER %BSY QLTH SEEK ACYL RATE CONN PEND DISCT
AMT036I B02 TSG002  45  .66   7 199  22   9   3  11
AMT038D -----
           ⑪  ⑫  ⑬
AMT038H  ASIDNAME ASID %BSY
AMT038I  WKR1      85  33
AMT038I  INIT      24  22
AMT038I  JES2      13  11
AMT038I  CMR8      77  33
    
```

Legend:

1. Device number whose activity was measured.
2. Volume serial number on the device measured.
3. Percentage of time an I/O request was active for the device.
4. Average number of I/O requests that were queued by the OS/390/XA I/O supervisor for the device.
5. Total number of head movements observed for the device during the interval (N/A for non-DASD and fixed-head devices).
6. Average number of cylinders moved for each seek observed for the device.
7. Number of I/O requests per second addressed to this device.
8. Average time (in milliseconds) the device was connected to a channel path on a per-request basis.
9. Average time (in milliseconds) each request was held, pending availability of a wait for path, device, or control unit.
10. Average time (in milliseconds) the device was disconnected from the channel path pending a device or control unit function (seek) on a per-request basis.
11. Name of an address space found to be a significant user of the device. In this field, ***** indicates that the address space terminated between the time the data was collected and the time the MDEVICE report was generated. *OTHERS* in this field indicates that the address space had more users during the monitoring period than could be listed. All activity for the overflow of users is reported as *OTHERS*.
12. Address space ID found to be a significant user of the device.
13. Percentage of the device busy time that this address space was requesting the device.

Usage Note

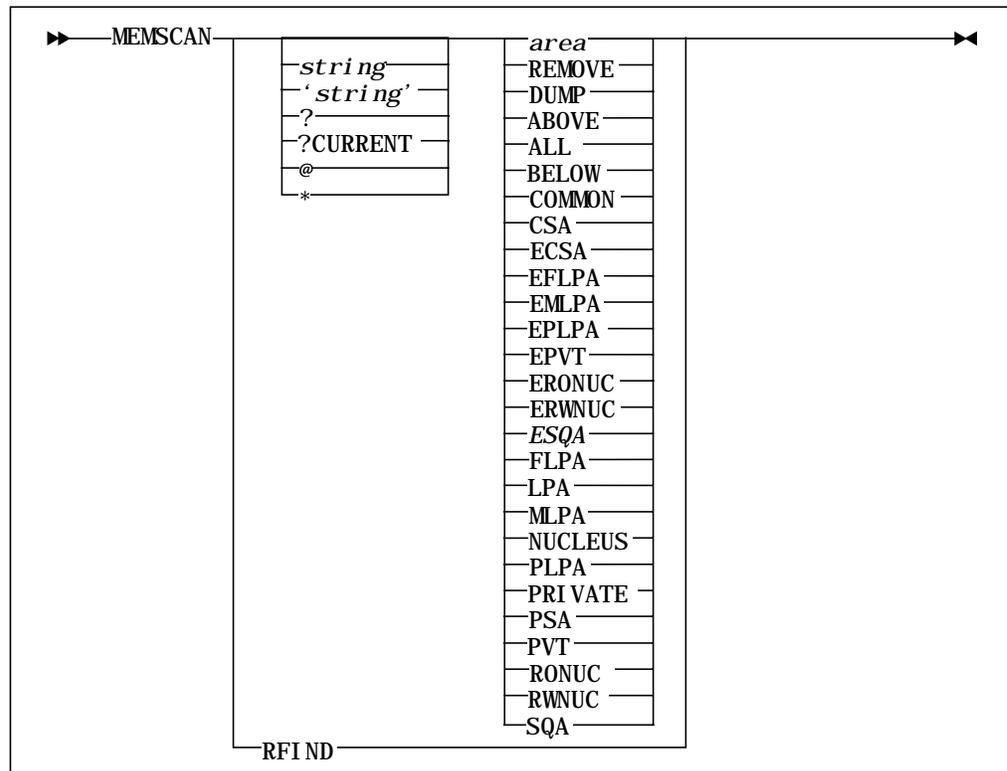
The MDEVICE service uses a sampling technique for measuring I/O activity. The default sampling period is 15 seconds at a rate of 4 samples per second. If you enter an override value for `time`, the sample rate is 4 samples per second if the period is 15 seconds or less and 2 samples per second if the period is greater than 15 seconds.

MEMSCAN

The MEMSCAN (ME) service scans the specified storage areas for the specified character or hexadecimal string and displays the location of each occurrence of the string. In addition, a symbol of the form @nn is created for each string occurrence until 99 symbols have been created. These symbols can then be used in conjunction with the DUMP and ZAP services to display or modify the located data.

When 99 symbols have been created, MEMSCAN displays 64 bytes of storage containing the next occurrence of the string and then terminates. The scan may be resumed to find the next occurrence by typing MEMSCAN without any operands. An asterisk (*) may be used to reference the address of the last occurrence of the string symbolically on the DUMP, EQUATE, and ZAP services.

Syntax



where

You may specify up to 10 parameters. The first parameter may be:

string Character strings that do not contain commas or blanks may be typed directly. Strings containing blanks or commands must be enclosed in single quotation marks (for example: ' ABC, DEF').

Note: Characters typed in lower case are converted to upper case. Type strings as hexadecimal data to search for lower case.

'string'	Hexadecimal data must be enclosed in single quotation marks and preceded by an X. For example: X' C1C2C36BC4C5C6' .
?	Displays a map of storage depicting the various areas within the address space.
@	Displays the symbols previously created by MEMSCAN.
*	Represents the address of the last location set by one of several services. Typing an asterisk is equivalent to entering the address as a hexadecimal string. See Example 1 on page 141.
Nul l	Omitting all parameters or omitting the first parameter causes MEMSCAN to resume an interrupted search.
area	Subsequent parameters (2nd through 10th) as described in the following section.
RFIND	Finds a previous scan.

Area Keywords

Any of the following keywords may be specified in any sequence, separated by commas or blanks. Only the first three letters of each keyword are required. Area keywords are cumulative in their effect. If an area keyword is not specified, the default is **COMMON**.

REMOVE	Causes any previously created symbols of the format @nn (where nn is 01 through 99) to be deleted prior to starting the scan.
DUMP	Causes MEMSCAN to display 64 bytes of storage surrounding each occurrence of the string.
ABOVE	Scans all storage above the 16-megabyte line.
ALL	Scans all storage (0 through 7FFFFFFF).
BELOW	Scans storage below the 16-megabyte line.
COMMON	Scans common storage, which comprises CSA, ECSA, SQA, and ESQA. The PSA, LPA, and nucleus area, although commonly addressee are excluded.
CSA	Scans the Common Service Area (CSA) located below the 16-megabyte line.
ECSA	Scans the Extended Common Service Area.
EFLPA	Scans the Extended Fixed Link-Pack Area.
EMLPA	Scans the Extended Modified Link-Pack Area.
EPLPA	Scans the Extended Pageable Link-Pack Area.
EPVT	Scans the Extended Private area.
	Note: Only the private area of the address space within which the SYSPROG Services service is executing is scanned.
ERONUC	Scans the Extended read-only nucleus.

ERWNUC	Scans the Extended read/write nucleus.
ESQA	Scans the Extended System Queue Area.
FLPA	Scans the Fixed Link-Pack Area.
LPA	Scans the entire Link-Pack Area. This includes FLPA, MLPA, PLPA, EFLPA, EMLPA, and EPLPA.
MLPA	Scans the Modified Link-Pack Area (below the 16-megabyte line).
NUCLEUS	Scans the entire nucleus. This includes RWNUC, RONU, ERWNUC, and ERONU.
PLPA	Scans the Pageable Link-Pack Area (below the 16-megabyte line).
PRIVATE	Scans private storage, both below and above the 16-megabyte line.
PSA	Scans the Prefixed Save Area.
PVT	Scans private storage below the 16-megabyte line.
RONUC	Scans the read-only nucleus below the 16-megabyte line.
RWNUC	Scans the read/write nucleus below the 16-megabyte line.
SQA	Scans the System Queue Area below the 16-megabyte line.

Examples

Example 1: To locate vectors to IGGPOST0:

```

lpa, iggpost0
AMTL41I MODULE=IGGPOST0 < EPLPA >
AMTL42I LA=030ADF38 EP=00AF72F8 (AMODE-31) LEN=000008
AMTL43I LPDE=00C43588
AMT001A RESOLVE PLUS
memscan, *
AMTVS8I SEARCHING FOR X' '
AMTVSWS WILL BE SCANNING: CSA 00800000-00B97FFF
AMTVSWS WILL BE SCANNING: SQA 00EBE000-00FC9FFF
AMTVSWS WILL BE SCANNING: EXTSQA 016D2000-020AFFFF
AMTVSWS WILL BE SCANNING: EXT-CSA 043D4000-07FFFFFFF
AMTVSNN SCANNING: CSA 00800000-00B97FFF
AMTVSNN SCANNING: SQA 00EBE000-00FC9FFF
AMTVS9I STRING FOUND AT 00C435AC IN SQA @01
AMTVSNN SCANNING: EXTSQA 016D2000-020AFFFF
AMTVSNN SCANNING: EXT-CSA 043D4000-07FFFFFFF
AMTVSSI SEARCH STOPPED AT THE END OF THE AREA (07FFFFFFF)
AMT001A RESOLVE PLUS
dump, @01-20
AMTC12I 00C4358C 00 00000000 C9C7C7D7 D6E2E3F0 80AF72F8 *... IGGPOST0... 8*
AMTC12I 00C4359C 00000000 00001800 B1220000 00000008 *.....*
AMTC12I 00C435AC 030ADF38 00C436C8 00000000 C9C6C7F0 *... D. H. ... IFGO*
AMTC12I 00C435BC F2F0F0D5 8261A458 00000000 00001800 *200Nb. u. ....*
    
```

Legend:

1. First, the LPA service is used to locate the start of IGGPOST0.
2. Then, MEMSCAN is used to search common storage for the address of IGGPOST0.
3. Finally, the DUMP service is used to display the storage area containing the located address constant.

Example 2: To produce a memory map, type:

```
mem ?
```

AMTVSH1	ALLOCATED STORAGE AREAS				
AMTVSH2	AREA TITLE	START	END	MEMSCAN	ABBR.
AMTVSLN	-----				
AMTVSAI	PSA	00000000	00001000	PSA	
AMTVSAI	PRI VATE	00001000	007FFFFFFF	PVT	
AMTVSAI	CSA	00800000	00B97FFF	CSA	
AMTVSAI	MLPA	00B98000	00C2FFFF	MLPA	
AMTVSAI	PLPA	00C30000	00EBDFFF	PLPA	
AMTVSAI	SQA	00EBE000	00FC9FFF	SQA	
AMTVSAI	NUC R/W	00FCA000	00FD9FFF	RWNUC	
AMTVSAI	NUC R/O	00FDA000	00FFFFFF	RONUC	
AMTVSAI	EXT- NUC R/O	01000000	01466FFF	ERONUC	
AMTVSAI	EXT- NUC R/W	01467000	016D1FFF	ERWNUC	
AMTVSAI	EXTSQA	016D2000	020AFFFF	ESQA	
AMTVSAI	EXT- PLPA	020B0000	043B9FFF	EPLPA	
AMTVSAI	EXT- FLPA	043BA000	043BCFFF	EFLPA	
AMTVSAI	EXT- MLPA	043BD000	043D3FFF	EMLPA	
AMTVSAI	EXT- CSA	043D4000	07FFFFFF	ECSA	
AMTVSAI	EXT- PRI VATE	08000000	7FFFFFFF	EPVT	

Example 3: To scan all storage for a hexadecimal string and display each occurrence, type:

```

mem, x'00af72f8', dump, all

AMTVS8I SEARCHING FOR X'00AF72F8'
AMTVSWS WILL BE SCANNING: ALL STORAGE
AMTVSNN SCANNING: PSA          00000000-00001000
AMTVSNN SCANNING: PRIVATE     00001000-007FFFFF
AMTVSNN SCANNING: CSA         00800000-00B97FFF
AMTVS9I STRING FOUND AT 00AF7384 IN CSA          @01
AMTVSLN -----
AMTVS5I 00AF7380 +04 F1000080 00AF72F8 00AF73B8 00AF73B8 *1      8      *
AMTVS5I 00AF7390      00000000 00000000 00000000 00000000 *          *
AMTVS5I 00AF73A0      00000000 00000000 00000000 00000000 *          *
AMTVS5I 00AF73B0      00000000 00000000 00AF9140 0751FE40 *          j      *
AMTVSNN SCANNING: MLPA          00B98000-00C2FFFF
AMTVSNN SCANNING: PLPA          00C30000-00EBDFFF
AMTVSNN SCANNING: SQA           00EBE000-00FC9FFF
AMTVSNN SCANNING: NUC R/W       00FCA000-00FD9FFF
AMTVSNN SCANNING: NUC R/O       00FDA000-00FFFFFF
AMTVSNN SCANNING: EXT-NUC R/O   01000000-01466FFF
AMTVSNN SCANNING: EXT-NUC R/W   01467000-016D1FFF
AMTVSNN SCANNING: EXTSQA        016D2000-020AFFFF
AMTVS9I STRING FOUND AT 01B2E0AC IN EXTSQA      @02
AMTVSLN -----
AMTVS5I 01B2E0A0 +0C 01B2FFA0 AD626E17 01BD9130 00AF72F8 *      > j      8*
AMTVS5I 01B2E0B0      00000080 8000A1E3 01B2E0A0 AD626E17 *      T : > *
AMTVS5I 01B2E0C0      01BD9130 00AF72B8 00000040 8000A0DB * j      *
AMTVS5I 01B2E0D0      01AFA1F0 AD6300B7 01A84958 07624020 * 0      y      *
AMTVSNN SCANNING: EXT-PLPA      020B0000-043B9FFF
AMTVSNN SCANNING: EXT-FLPA      043BA000-043BCFFF
AMTVSNN SCANNING: EXT-MLPA      043BD000-043D3FFF
AMTVSNN SCANNING: EXT-CSA       043D4000-07FFFFFFF
AMTVSNN SCANNING: EXT-PRIVATE   08000000-7FFFFFFF
AMTVSSI SEARCH STOPPED AT THE END OF THE AREA (7FFFFFFF)
    
```

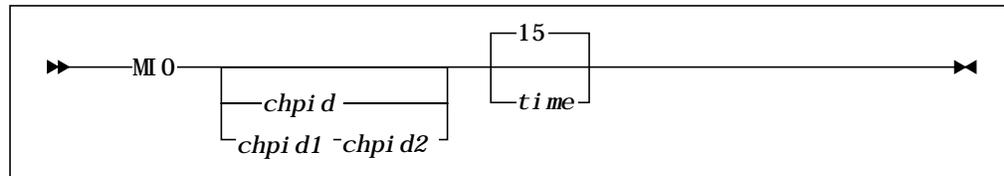
Usage Notes

- You can omit previously typed parameters if you do not want to change them. For example, if you had previously typed MEMSCAN TEXT PRIVATE (which searches for the string TEXT in private storage), you can search for the string TEXT in common storage by typing MEMSCAN, , COMMON.
- The ATTENTION key interrupts MEMSCAN and saves all current values. The scan can be resumed at the point of interruption by typing MEMSCAN REFIN D or just MEMSCAN.
- Do not embed blanks in the search string.

MIO

The MIO service monitors the overall performance of the entire I/O subsystem, or a subset thereof, to detect I/O bottlenecks causing workload delays. This service provides workload (address space-related) information, which lets you detect jobs monopolizing specific devices and channel paths.

Syntax



where

chpi d Is a channel path identifier.

chpi d1 - chpi d2 Is a range of channel path identifiers; *chpi d1* is the lowest channel path identifier and *chpi d2* is the highest channel path identifier.

time Specifies the length of the sample period in seconds. The default sample period is 15 seconds.

Note: If you do not specify *chpi d*, the MIO service monitors all channel paths attached to all processors.

Examples

To display information on all channel paths, type:

```

mi o

AMT030I I/O DATA BEING GATHERED

AMT031I CHANNEL PATH UTILIZATION REPORT
          ①    ②    ③
AMT032I CHANNEL PATH 00 (BYTE) WAS 5% BUSY
AMT032I CHANNEL PATH 03 (BLOCK) WAS 23% BUSY
AMT032I CHANNEL PATH 12 (BLOCK) WAS 45% BUSY
AMT032I CHANNEL PATH 23 (BLOCK) WAS 17% BUSY

AMT033I PROCESSOR I/O INTERRUPT ACTIVITY REPORT
          ④    ⑤    ⑥
AMT034I CPU 00 PROCESSED 3187 INTERRUPTS ( 12% WERE THROUGH TPI)
AMT034I CPU 02 PROCESSED 110 INTERRUPTS ( 11% WERE THROUGH TPI)

AMT035I DEVICE AND LOGICAL CONTROL UNIT REPORT
          ⑦    ⑧    ⑨    ⑩    ⑪    ⑫    ⑬    ⑭    ⑮    ⑯
AMT036D -----
AMT036H DEV VOLSER %BSY QLTH SEEK ACYL RATE CONN PEND DISCT
AMT036I 262 SPOOL1 20 .34 180 149 6 2 4 15
AMT038D -----
          ⑰    ⑱    ⑲
AMT038H ASI DNAME ASID %BSY
AMT038I WKR1 85 33
AMT038I INIT 24 22
AMT038I JES2 13 11
AMT038I CMR8 77 33
AMT038I ***** 54 2
AMT038I DAD2 70 4
AMT038I *OTHERS* 96 2
AMT038I *OTHERS* 96 2
          ⑳    ㉑    ㉒    ㉓    ㉔    ㉕
AMT039I LCU 07: I/O RATE 5/SEC %OK 73 %DEFER 14/ 38 AQD 1
    
```

Legend:

1. Channel path identifier (CHPID).
2. Type of channel path as specified during IOCP generation. The valid path types are BLOCK, for block multiplexing paths, and BYTE, for byte multiplexing paths.
3. Percentage of time the channel path was busy during the sample period.

Note: Channel paths less than 1% busy during the sample period are not reported unless specified either explicitly or implicitly (using a path range) as an operand of the MIO service.

4. CPU identifier (CPUID) for the processor whose activity is being reported.
5. Total number of I/O interrupts (including TPIs) processed by this CPU in the sample period.
6. Percentage of I/O interrupts processed by this CPU that did not require a physical PSW swap (that is, were processed through the Test Pending Interrupt (TPI) instruction).

7. Device number whose activity was measured.
8. Volume serial number of the device measured.
9. Percentage of time an I/O request was active for the device.
10. Average number of I/O requests queued by the I/O supervisor for the device.
11. Total number of head movements observed for the device during the sample period (N/A for non-DASD and fixed-head devices).
12. Average number of cylinders moved for each seek observed for the device.
13. Number of I/O requests per second addressed to this device.
14. Average time (in milliseconds) the device was connected to a channel path on a per-request basis.
15. Average time (in milliseconds) each request was held, pending availability of a waiting for path, device, or control unit.
16. Average time (in milliseconds) the device was disconnected from the channel path pending a device or control unit function (seek) on a per-request basis.
17. Name of an address space found to be a significant user of the device. In this field, ***** indicates that the address space terminated between the time the data was collected and the time the MDEVICE report was generated. *OTHERS* in this field indicates that the address space had more users during the sample period than could be listed. All activity for the overflow of users is reported as *OTHERS*.
18. Number of the address space found to be a significant user of the device.
19. Percentage of the device busy time that this address space was requesting the device.
20. Number of the logical control unit with which all the devices reported above (back to the previous LCU report line) are associated.
21. Total number of I/O requests per second received for all devices associated with the LCU.
22. Percentage of I/O requests processed by this LCU initiated successfully.
23. Percentage of I/O requests processed by this LCU that were deferred because of a busy physical control unit.
24. Percentage of I/O requests processed by this LCU that were deferred because of a busy physical device.
25. Average number of I/O requests held (queued) by this LCU pending channel path, physical control unit, or device availability.

Usage Note

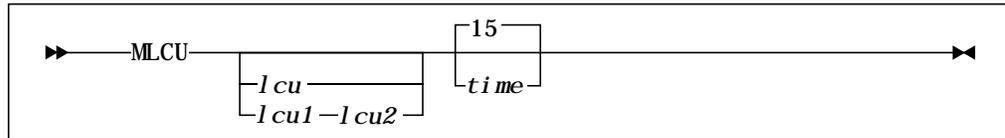
The MIO service uses a sampling technique for measuring I/O activity. The default sampling period is 15 seconds at a rate of 4 samples per second. If you enter an override value for `time`, the sample rate is 4 samples per second if the period is 15 seconds or less and 2 samples per second if the period is greater than 15 seconds.

MLCU

The MLCU (MLC) service monitors the performance and utilization of logical control units configured into the I/O subsystem. The service displays data (I/O rate, requests deferred for control unit, and device busy conditions) for logical control units that have I/O activity during the monitoring period.

Use this service to determine the optimal physical device placement and to evaluate shared DASD contention.

Syntax



where

- l cu* Is a logical control unit.
- l cu1-l cu2* Is a range of logical control units; *l cu1* is the lowest-numbered logical control unit and *l cu2* is the highest-numbered logical control unit.
- time* Specifies the length of the sample period in seconds. The default sample period is 15 seconds.

Example 1 (for Processors Older than 3090)

To monitor all LCUs for 15 seconds, type:

```
ml cu
AMT040I LOGICAL CONTROL UNIT DATA BEING GATHERED
AMT041I LOGICAL CONTROL UNIT ACTIVITY REPORT
      ①          ②          ③          ④ ⑤          ⑥
AMT039I LCU 03: I/O RATE 5/SEC %OK 87 %DEFER 2/ 11 AQP 1
AMT039I LCU 07: I/O RATE 3/SEC %OK 93 %DEFER 5/ 2  AQP 1
AMT039I LCU 1A: I/O RATE 23/SEC %OK 81 %DEFER 9/ 10 AQP 3
AMT039I LCU 21: I/O RATE 48/SEC %OK 44 %DEFER 39/ 17 AQP 5
AMT039I LCU 2F: I/O RATE 9/SEC %OK 73 %DEFER 14/ 13 AQP 2
```

Legend:

1. Logical control unit number.
2. Total number of I/O requests/second received for devices associated with the LCU.
3. Percentage of I/O requests processed by this LCU that were initiated successfully.
4. Percentage of I/O requests deferred because of a busy physical control unit.

5. Percentage of I/O requests processed by this LCU that were deferred because of a physical device busy condition.
6. Average number of I/O requests that were held (queued) by this LCU pending channel path, physical control unit, or device availability.

Example 2 (for Processors Older than 3090)

To monitor LCUs from 0 to 2E for 5 seconds, type:

```
ml cu 000-02E 5

AMT040I LOGICAL CONTROL UNIT DATA BEING GATHERED

AMT041I LOGICAL CONTROL UNIT ACTIVITY REPORT
AMT063I IOP 00: ACTIVITY RATE 512   AVG Q LENGTH .02
AMT063I IOP 01: ACTIVITY RATE .00   AVG Q LENGTH .00
AMT061I LCU 0013 CONTENTION RATE    2   ALL PATHS BUSY    0 %SWITCH BUSY  0
AMT062I LCU 0013 DELAY Q LENGTH    .66   CHPID TAKEN      1 %CU BUSY    37
AMT061I LCU 001A CONTENTION RATE    .00   ALL PATHS BUSY    0 %SWITCH BUSY  0
AMT062I LCU 001A DELAY Q LENGTH    .00   CHPID TAKEN      .60 %CU BUSY    0
AMT061I LCU 001D CONTENTION RATE    .00   ALL PATHS BUSY   20 %SWITCH BUSY  0
AMT062I LCU 001D DELAY Q LENGTH    .00   CHPID TAKEN      21 %CU BUSY    0
AMT061I LCU 0024 CONTENTION RATE    .00   ALL PATHS BUSY    0 %SWITCH BUSY  0
AMT062I LCU 0024 DELAY Q LENGTH    .00   CHPID TAKEN      59 %CU BUSY    0
AMT061I LCU 0029 CONTENTION RATE    .00   ALL PATHS BUSY    0 %SWITCH BUSY  3
AMT062I LCU 0029 DELAY Q LENGTH    .00   CHPID TAKEN     111 %CU BUSY    2
AMT061I LCU 002E CONTENTION RATE    .00   ALL PATHS BUSY    0 %SWITCH BUSY  0
AMT062I LCU 002E DELAY Q LENGTH    .00   CHPID TAKEN     129 %CU BUSY    0
```

Example 3 (for 3090 and Newer Processors)

To monitor all LCUs for 15 seconds, type:

```
ml cu

AMT040I LOGICAL CONTROL UNIT DATA BEING GATHERED

AMT041I LOGICAL CONTROL UNIT ACTIVITY REPORT
AMT063I IOP 01: ACTIVITY RATE 10000   AVG Q LENGTH 20000
AMT061I LCU 023: CONTENTION RATE 30005   ALL PATHS BUSY 40001
AMT062I LCU 023: DELAY Q LENGTH 50000   CHPID TAKEN 60000   % CU BUSY 70000
```

Legend:

1. Total number of I/O requests per second received for all devices associated with the LCU.
2. Average depth of the I/O request queue.
3. Rate at which I/Os were queued on the LCU in the I/O subsystem.
4. Percentage of time that all paths to a given I/O device on that LCU were busy.
5. Average length of the queue for the LCU.

6. Percentage of I/O requests processed by this LCU that were initiated successfully.
7. Percentage of I/O requests processed by this LCU that were deferred because of a busy physical control unit.

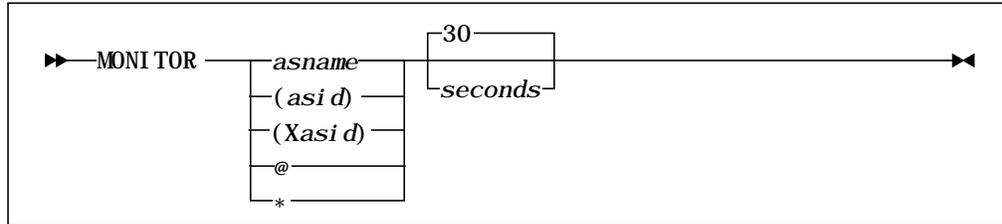
Usage Note

The MLCU service uses a sampling technique for measuring I/O activity. The default sampling period is 15 seconds at a rate of 4 samples per second. If you enter an override value for `time`, the sample rate is 4 samples per second if the period is 15 seconds or less and 2 samples per second if the period is greater than 15 seconds.

MONITOR

The MONITOR (MON) service monitors address space activity to determine whether an address space is in a wait or a looping state. The service provides the basis for a deadline management system by providing CPU time and EXCP counts for the life of a step. For a job whose characteristics are known, you can gauge the progress of a step at any point during its execution.

Syntax



where

<i>asname</i>	Is the address space name.
<i>(asi d)</i>	Is the address space identifier in decimal format.
<i>(Xasi d)</i>	Is the address space identifier in hexadecimal format.
@	Specifies your own address space.
*	Specifies the last address space entered.
<i>seconds</i>	Is the number of seconds the address space is to be monitored; the default is 30 seconds.

Example

To monitor address space INVENTORY for 30 seconds, type:

```

monitor inventory

AMTJ22I STATISTICS BEING GATHERED FOR INVENTORY
AMTJ21I ① JOB 295 ② INVENTORY ③ STEP1 ④ PRTY EE(238) ⑤ PGP 22/2
AMTJ23I LIFE OF STEP TOTAL CPU 26.33 EXCP 1233 PAGES 8 SUN 10300
AMTJ24I LAST 30 SECOND CPU ⑦ 3.95 EXCP ⑧ 70 PAGES ⑨ 0 SUN ⑩ 500
  
```

Legend:

1. JES job ID.
2. Address space name.
3. Current stepname.

4. Priority.
5. Performance group.
6. Performance period.
7. CPU time for the life of the step during the time specified.
8. Count of EXCPs for the life of the step during the time specified.
9. Total number of page actions during the step during the time specified.
10. Total service units for the step during the time specified.

MOUNT

The MOUNT (MOU) service mounts all online direct access storage devices that are not allocated or mounted as public or storage volumes. This service also provides a shorthand method for issuing mount commands for removable disk devices, as opposed to issuing the OS/390 MOUNT command for each volume.

Syntax

```
▶▶ MOUNT ◀◀
```

Example

To mount all online direct access storage devices, type:

```
mount
AMTM11I   ①  PACK01  MOUNTED  ON  23A  (③  PRIVATE/RESERVED)
AMTM12I   ⑤  1  DIRECT ACCESS VOLUME(S) MOUNTED
```

Legend:

1. Volume serial of the disk device mounted.
2. Device number.
3. Volume-use attribute (PRIVATE/PUBLIC/STORAGE).
4. Volume-mount attribute (RESERVED/RESIDENT).
5. Number of devices whose mount or use attributes were changed by service MOUNT. If no devices were affected, this field contains a 0 (zero).

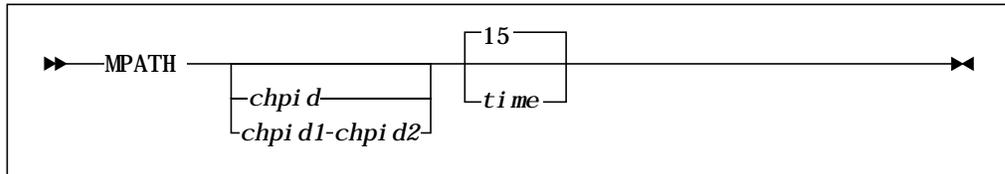
Usage Notes

- The MOUNT service affects every online DASD (Direct Access Storage Device) that:
 - Is not currently allocated
 - Is not in the process of being taken offline
 - Does not contain an active page data set
 - Is not already marked either RESIDENT or RESERVED
- The MOUNT service marks all DASD except those marked PUBLIC and REMOVABLE as mounted in the UCB (that is, with mount status =RESERVED). It does not do physical I/O to the pack and does not fill in volume serial information in the UCB.

MPATH

The MPATH (MPA) service monitors I/O channel path activity. This service also detects imbalances and possible path failures by monitoring the utilization of all, or a subset of, the channel paths installed in the system.

Syntax



where

chpi d Is a channel path identifier.

chpi d1- chpi d2 Is a range of channel path identifiers; *chpi d1* is the lowest channel path identifier and *chpi d2* is the highest channel path identifier.

time Specifies the length of the sample period in seconds. The default sample period is 15 seconds.

Note: If you do not specify *chpi d*, the MPATH service monitors all channel paths attached to all processors.

Example

To display I/O activity for all channel paths, type:

```

mpath
      ①           ②
AMT021I CHANNEL PATH 0C WAS 4% BUSY
AMT021I CHANNEL PATH 12 WAS 1% BUSY
AMT021I CHANNEL PATH 13 WAS 1% BUSY
AMT021I CHANNEL PATH 1B WAS 13% BUSY
AMT021I CHANNEL PATH 1C WAS 7% BUSY
AMT021I CHANNEL PATH 38 WAS 14% BUSY
AMT021I CHANNEL PATH 39 WAS 19% BUSY
AMT021I CHANNEL PATH 3C WAS 19% BUSY
AMT021I CHANNEL PATH 3D WAS 11% BUSY
    
```

To display I/O activity for channel paths 12–14, type:

```

mpath 12-14
      ①           ②
AMT021I CHANNEL PATH 12 WAS 4% BUSY
AMT021I CHANNEL PATH 13 WAS 3% BUSY
AMT021I CHANNEL PATH 14 WAS 0% BUSY
    
```

To display I/O activity for all channel paths over the past 60 seconds, type:

```
mpa, , 60
      ①      ②
AMT021I CHANNEL PATH 0C WAS 4% BUSY
AMT021I CHANNEL PATH 12 WAS 3% BUSY
AMT021I CHANNEL PATH 13 WAS 3% BUSY
AMT021I CHANNEL PATH 14 WAS 4% BUSY
AMT021I CHANNEL PATH 1B WAS 5% BUSY
AMT021I CHANNEL PATH 1C WAS 5% BUSY
AMT021I CHANNEL PATH 1E WAS 4% BUSY
AMT021I CHANNEL PATH 38 WAS 5% BUSY
AMT021I CHANNEL PATH 39 WAS 11% BUSY
AMT021I CHANNEL PATH 3C WAS 11% BUSY
AMT021I CHANNEL PATH 3D WA 6% BUSY
```

Legend:

1. Channel path identifier (CHPID).
2. Percentage of time the channel path was busy during the sample period.

Usage Note

The MPATH service uses a sampling technique for measuring I/O activity. The default sampling period is 15 seconds at a rate of 10 samples per second. If you enter an override value for `time`, the sample rate is 10 samples per second if the period is 15 seconds or less and 5 samples per second if the period is greater than 15 seconds.

MSTORAGE

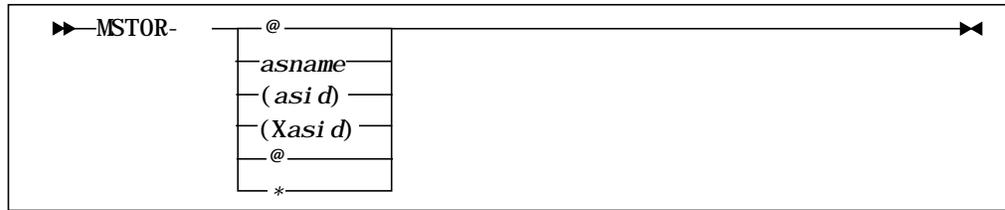
The MSTORAGE (MS) service maps virtual storage areas above and below the 16MB line, including location, length and percentage used when applicable. Areas common to all address spaces and private area locations specific to the specified address space are displayed, when available.

The private area above and below the 16MB line is divided into three sections:

- LSQA/SWA—allocated from the top of the private area down.
- User committed private area—allocated from the bottom up.
- Uncommitted area—between the LSQA/SWA and user committed private area.

The current utilization within the LSQA/SWA and committed private area is displayed. LSQA/SWA may expand into the uncommitted area as needed. Committed private area may expand into the uncommitted private area to the user limit value.

Syntax



where

- asname* Is the address space name.
- (asi d)* Is the address space identifier in decimal format.
- (Xasi d)* Is the address space identifier in hexadecimal format.
- @ Specifies your own address space.
- * Specifies the last address space entered.

Example

To display the virtual storage areas for BMVDWP3, type:

AMTMS0I Address space: BMVDWP3 ①						
AMTMS0I	AREA ②	START ③	END ④	LENGTH (K) ⑤	%USED ⑥	USER LMT ⑦
AMTMS2I	-----	-----	-----	-----	-----	-----
AMTMS3I	Ext LSQA/SWA	7F6C8000	7FFFFFFF	9,440K	99	
AMTMS3I	Uncommitted Ext-Pvt	11A8A000	7F6C7FFF	1,798,392K	---	139FFFFFF
AMTMS3I	Committed Ext-Pvt	11A00000	11A89FFF	552K	77	
AMTMS3I	Ext CSA	051D0000	119FFFFFF	204,992K	28	
AMTMS3I	Ext MLPA	051BA000	051CFFFF	88K	---	
AMTMS3I	Ext FLPA	051B7000	051B9FFF	12K	---	
AMTMS3I	Ext PLPA	0271A000	051B6FFF	43,636K	---	
AMTMS3I	Ext SQA	01898000	02719FFF	14,856K	82	
AMTMS3I	Ext NUC R/W	01596000	01897FFF	3,080K	---	
AMTMS3I	Ext NUC R/O	01000000	01595FFF	5,720K	---	
AMTMS4I	16 MEG LINE -----	-----	-----	-----	-----	-----
AMTMS3I	NUC R/O	00FDB000	00FFFFFF	148K	---	
AMTMS3I	NUC R/W	00FC9000	00FDAFFF	72K	---	
AMTMS3I	SQA	00EBA000	00FC8FFF	1,084K	50	
AMTMS3I	PLPA	00C0D000	00EB9FFF	2,740K	---	
AMTMS3I	MLPA	00B8A000	00C0CFFF	524K	---	
AMTMS3I	CSA	00800000	00B89FFF	3,624K	93	
AMTMS3I	LSQA/SWA	0075A000	007FFFFFF	664K	54	
AMTMS3I	Uncommitted Private	0005F000	00759FFF	7,148K	---	00414FFF
AMTMS3I	Committed Private	00005000	0005EFFF	360K	104	
AMTMS3I	V=R (IF ANY)	00005000	00024FFF	128K	---	
AMTMS3I	Syste					

Legend:

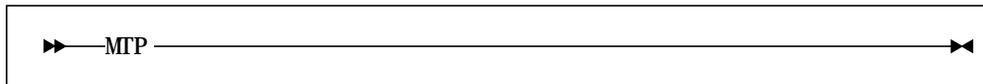
1. Name of the address space.
2. Name of the virtual storage area.
3. Lowest virtual storage address for this storage area.
4. Highest virtual storage address for this storage area.
5. Number of kilobytes contained in this storage area.
6. Percent of the area in use, where available.
7. Region limit value.

MTP

The MTP service

- Displays volume serial numbers, device numbers, device types, and address space names for tape and direct access storage devices that have pending mount requests.
- Identifies the source of the malfunction when an address space stops processing.
- Finds which address space requests a particular volume.

Syntax



Example

To identify pending mount requests, type:

```

mtp
AMTM22I      MOUNT PENDING FOR B90078 UNIT= 384 (3400- 4) JOB= GJJSRUN

```

Legend:

1. Volume serial number of the requested volume.
2. Number of the device that requires the volume mount.
3. Device type.
4. Name of the address space requesting the unmounted volume.

Note: Only the volume serial number requested by the allocation appears in message AMTM22I. The second and successive volumes of a data set appear as blanks in message AMTM22I.

To display the entry-point address for hex location 01224018, type:

```
nucleus 1224018 loc
          ①          ②          ③
AMTN12I LOC 01224018 IS + 000018 FROM ENTRY IECROUTE
AMTN13I EPA=01224000 (AMODE- 31) LTH=000780
```

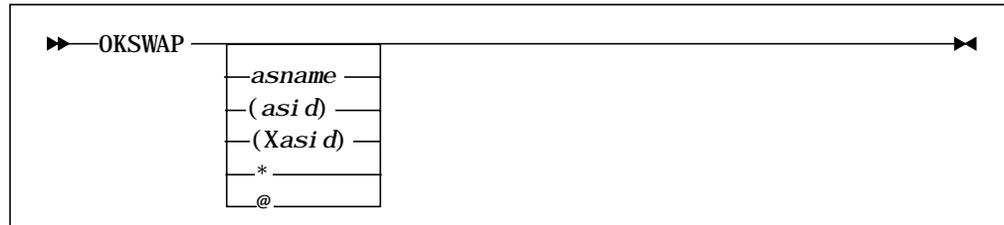
Legend:

1. Requested hex location.
2. Hexadecimal offset of the hex location from the nucleus entry point that contains the hex location.
3. Name of the closest nucleus entry point to the requested hex location.

OKSWAP

The OKSWAP (OKS) service reverses the effects of the DONTSWAP service, making an address space swappable.

Syntax



where

@ Specifies your own address space.

asname Is the address space name.

(*asi d*) Is the address space identifier in decimal format.

(*Xasi d*) Is the address space identifier in hexadecimal format.

* Specifies the last address space entered.

Note: If you do not specify an address space to be made swappable, you are prompted to define an address space.

Examples

To make address space INVENTORY swappable, type:

```

okswap inventory
AMTS61I   STC 6 INVENTORY STEP3 PRTY 7(116) PGP 2/3
AMTS62I   INVENTORY HAS BEEN SET SWAPPABLE (00)
  
```

To reduce the nonswappability count of address space INVENTORY, type:

```

okswap inventory
AMTS61I   STC 6 INVENTORY STEP3 PRTY 7(116) PGP 2/3
AMTS64I   INVENTORY NONSWAPPABILITY COUNT REDUCED (02)
  
```

Legend:

1. JES job ID, address space name, and current stepname of the specified address space.

2. External and internal dispatching priority of the address space.
3. Performance group and performance group period.
4. Nonswappability count. The operating system keeps this count for the address space. If the count is greater than zero, the address space is still nonswappable. The OKSWAP service reduces the count by one. When the count reaches zero, the address space is swappable.

Usage Note

The OKSWAP service reverses the effects of the DONTSWAP service. Using the OKSWAP service to make system-specified nonswappable address spaces swappable is not recommended.

PAGING

The PAGING (PAG) service is the paging monitor for the OS/390 system. This service provides the total paging rate and divides the total paging rate into its components for analysis. It also provides the page reclaim rate.

Syntax

```
▶▶ PAGING ◀◀
```

Example

To display paging information, type:

```

pagi ng
AMTP11I  TOTAL PAGING RATE  26. 75 PAGES/SEC FOR LAST 65. 25 SECONDS
AMTP12I  DEMAND PAGING      10. 05 PAGES/SEC
AMTP13I  VIO PAGING RATE   3. 60 PAGES/SEC
AMTP14I  SWAP PAGING RATE  9. 75 PAGES/SEC
AMTP15I  LPA PAGING RATE   3. 79 PAGES/SEC
AMTP16I  CSA PAGING RATE   6. 44 PAGES/SEC
AMTP17I  PAGE RECLAIM RATE  2. 50 PAGES/SEC
  
```

Legend:

1. Total system paging rate over the last data collection interval; total system paging is the sum of VIO paging operations (in and out), swapping operations, and all other paging operations (excluding CSA paging and page reclaims).
2. Duration of the last data collection interval.
3. Paging rate caused by virtual I/O.
4. Paging rate caused by swapping.
5. Paging rate in the system link-pack area (LPA).
6. Paging rate in the common service area (CSA).
7. Page reclaim rate.

Usage Note

The period for which paging data is reported may vary, based on the time the paging activity counters were last cleared.

PARMLIST

The PARMLIST (PAR, PLIST, or PL) service lists the contents of SYS1.PARMLIB members. This service helps you examine system initialization and operation parameters.

Syntax

```
▶▶ PARMLIST member ▶▶
```

where

member Is a member in SYS1.PARMLIB.

Example

To list the contents of SYS1.PARMLIB member IEASYS00, type:

```
parmlist ieasys00

AMTCF11  APF=00,
AMTCF11  CMD=00,
AMTCF11  CSA=3072,
AMTCF11  CVI 0,
AMTCF11  FIX=00,
AMTCF11  ICS=00,
AMTCF11  IOS=00,
AMTCF11  IPS=00,
AMTCF11  LNK=00,
AMTCF11  MLPA=(00),
AMTCF11  LOGCLS=L,
AMTCF11  LOGLMT=050000,
AMTCF11  MAXUSER=50,
AMTCF11  OPT=00,
AMTCF11  PAGE=(SYS1. PAGEA,
AMTCF11           SYS1. LOCALA,
AMTCF11           SYS1. LOCALB,
AMTCF11           SYS1. LOCALC,
AMTCF11           SYS1. LOCALD),
AMTCF11  PAGNUM=(4, 3),
AMTCF11  REAL=192,
AMTCF11  RSU=2,
AMTCF11  SMF=00,
AMTCF11  SQA=8,
AMTCF11  SWAP=(SYS1. SWAPA
AMTCF11           SYS1. SWAPB),
AMTCF11  VAL=00,
AMTCF11  VRREGN=192,
```

PIO

The PIO (PI) service monitors I/O interrupt activity and determines the distribution of I/O interrupts among processors.

Syntax



where

time Specifies the length of the sample period in seconds. The default sample period is 15 seconds.

Example

To display I/O interrupt activity, type:

```

pio

AMT070I I/O INTERRUPT DATA BEING GATHERED

AMT073I PROCESSOR I/O INTERRUPT ACTIVITY REPORT
AMT074I CPU 00 PROCESSED      300 INTERRUPTS (  2% WERE THROUGH TPI)
AMT074I CPU 02 PROCESSED     1387 INTERRUPTS ( 11% WERE THROUGH TPI)

```

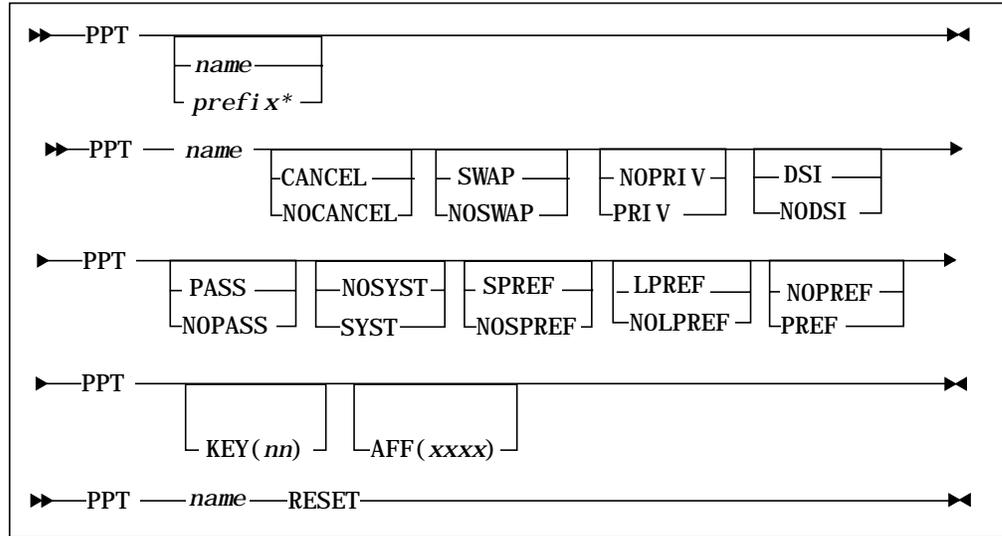
Legend:

1. CPU identifier (CPUID) for the processor whose activity is being reported.
2. Total number of I/O interrupts (including TPIs) processed by this CPU in the interval.
3. Percentage of I/O interrupts processed by this CPU through the Test Pending Interrupt (TPI) instruction (the percentage of I/O interrupts that do not require a physical program status word (PSW) swap).

PPT

The PPT service displays and updates the program properties table (PPT).

Syntax



Description

The PPT service is used to

- Display the entire program properties table (PPT).
- Display the PPT entry for all programs that begin with a string of one or more characters.
- Display the PPT entry for a specific program.
- Functionally delete a PPT entry (remove all special attributes).
- Add or change PPT entries. The presence of one or more keywords in addition to the program name indicates that:
 - A new entry will be created, if one does not currently exist.
 - If an entry exists, it should be updated according to the keywords that were specified. Only the specified attributes are changed.

The following table lists the supported bimodal keywords. The first column represents the IBM default value, and the second column represents the opposite meaning from the value in the first column. If neither keyword in a row is specified, the existing value is retained. When a new entry is created, all attributes are set to the values in the first column prior to applying the specified keywords. Keywords that begin with NO can be abbreviated to their first 5 characters. Other keywords can be abbreviated to their first 3 three characters.

Column 1	Column 2
CANCEL	NOCANCEL
SWAP	NOSWAP
NOPRIV	PRIV
DSI	NODSI
PASS	NOPASS
NOSYSTAS	SYSTASK
NOLPREF or NO1P	LPREF or 1P
NOSPREF or NO2P	SPREF or 2P
PREF	NOPREF

Other keyword parameters:

Keyword	Description
KEY(<i>nn</i>)	Where <i>nn</i> is 0-15. (The default is 8.)
AFF(<i>xxxx</i>)	Where <i>xxxx</i> is a four-character hexadecimal mask representing the processors upon which the program may execute. The default is AFF(FFFF). CPU is an alias for AFF.
RESET	When specified indicates that all IBM defaults are to be applied to the entry (all special attributes removed). RESET can not be specified in conjunction with any other keyword.

Enter	To
<code>ppt</code>	Display all entries in the PPT.
<code>ppt, prefix*</code>	Display all PPT entries that begin with the <i>prefix</i> .
<code>ppt, name</code>	Display the PPT entry for a program <i>name</i> .
<code>ppt, name, reset</code>	Reset all of the attributes in the PPT entry for the program <i>name</i> to the IBM defaults. This effectively deletes the entry.
<code>ppt, name, attr1, attr2, . . . attrn</code>	Create a PPT entry for the program <i>name</i> if one does not already exist and sets the specified attributes. Unspecified attributes remain as currently defined. For example: <code>PPT name, NOSWAP,CANCEL</code> would add the NOSWAP attribute, remove the NOCANCEL attribute, and leave all others unchanged.
<code>ppt, name, AFF(xxx)</code>	Set the CPU affinity mask for the PPT entry for program <i>name</i> .
	Note: All four hexadecimal characters in the mask must be specified.

Example 1

To display all the PPT entries, type:

```

ppt
AMTTPP1I Program Non Non SYS DS Bypass CPU Storage Pref
AMTTPP2I Name Cancel Swap Key Priv Task Integ Password Mask
AMTTPP3I -----
AMTTPP4I IEDQTCAM No Yes 6 No No Yes No FFFF NoPref
AMTTPP4I I STINM01 Yes Yes 6 No Yes Yes Yes FFFF NoPref
AMTTPP4I I KTCAS00 Yes No 6 Yes Yes Yes No FFFF
AMTTPP4I AHLGTF Yes Yes 0 No Yes Yes No FFFF NoPref
AMTTPP4I HHLGTF Yes Yes 0 No Yes Yes No FFFF NoPref
AMTTPP4I IHLGTF Yes Yes 0 No Yes Yes No FFFF NoPref
AMTTPP4I IEFIIC Yes No 0 Yes Yes Yes No FFFF
AMTTPP4I IEEMB860 Yes Yes 0 No Yes No Yes FFFF
AMTTPP4I IEEVMNT2 Yes No 0 No Yes Yes No FFFF
AMTTPP4I IASXW00 Yes No 1 No Yes Yes No FFFF
AMTTPP4I CSVVFCRE No Yes 0 No Yes Yes No FFFF
AMTTPP4I HASJES20 Yes Yes 1 No Yes No No FFFF
AMTTPP4I DFSMVRCO No Yes 7 No Yes Yes No FFFF NoPref
AMTTPP4I IATINTK Yes Yes 1 No Yes No No FFFF
AMTTPP4I DXRRLM00 No Yes 7 No Yes Yes No FFFF NoPref
AMTTPP4I APSPIEP No Yes 1 No Yes No No FFFF NoPref
AMTTPP4I AKPCSEIP No Yes 1 No Yes No No FFFF NoPref
AMTTPP4I IATINTKF No Yes 1 No Yes No No FFFF
AMTTPP4I DSNUTILB No No 7 No No Yes No FFFF
AMTTPP4I IEAVTDSV No Yes 0 No Yes Yes No FFFF
AMTTPP4I IFASMF Yes Yes 0 Yes Yes No No FFFF
AMTTPP4I CSVLLCRE No Yes 0 No Yes Yes Yes FFFF
AMTTPP4I AVFMNBLD Yes Yes 3 No Yes Yes No FFFF NoPref
AMTTPP4I ERBMFMFC No Yes 8 No Yes No No FFFF
AMTTPP4I ERB3GMFC No Yes 8 No Yes No No FFFF
AMTTPP4I IGGCLX0 Yes Yes 0 Yes Yes No No FFFF SP
AMTTPP4I IGDSI01 Yes Yes 5 No Yes Yes Yes FFFF
    
```

Example 2

To display all PPT entries beginning with **IEDQ**, type:

```

ppt, i edq*
AMTTPP1I Program Non Non SYS DS Bypass CPU Storage Pref
AMTTPP2I Name Cancel Swap Key Priv Task Integ Password Mask
AMTTPP3I -----
AMTTPP4I IEDQTCAM No Yes 6 No No Yes No FFFF NoPref
    
```

Example 3

To reset all attributes for program name DWPTEST, if it exists in the table, type:

```
ppt, dwptest, reset

AMTPP9I Add/Update successful.
AMTPP0I
AMTPP1I Program Non Non SYS DS Bypass CPU Storage Pref
AMTPP2I Name Cancel Swap Key Priv Task Integ Password Mask
AMTPP3I -----
AMTPP4I DWPTEST No No 8 No No Yes No FFFF
```

Note: If DWPTEST does not currently exist, it is created.

Example 4

To set two attributes in the entry for DWPTEST, if it exists in the table, type:

```
ppt, dwptest, nocan, noswap

AMTPP9I Add/Update successful.
AMTPP0I
AMTPP1I Program Non Non SYS DS Bypass CPU Storage Pref
AMTPP2I Name Cancel Swap Key Priv Task Integ Password Mask
AMTPP3I -----
AMTPP4I DWPTEST Yes Yes 8 No No Yes No FFFF
```

Note: If DWPTEST does not currently exist, it is created.

Example 5

To change the key for entry DWPTEST, if it exists, type:

```
ppt, dwptest, key(4)

AMTPP9I Add/Update successful.
AMTPP0I
AMTPP1I Program Non Non SYS DS Bypass CPU Storage Pref
AMTPP2I Name Cancel Swap Key Priv Task Integ Password Mask
AMTPP3I -----
AMTPP4I DWPTEST Yes Yes 4 No No Yes No FFFF
```

Note: If DWPTEST does not currently exist, it is created.

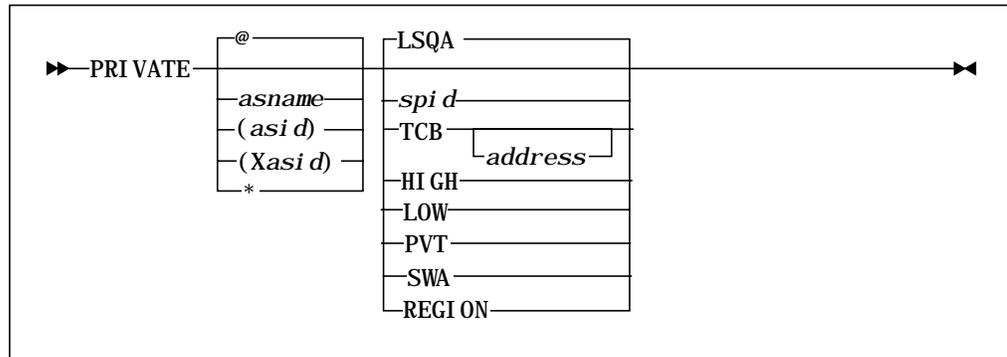
PRIVATE

The PRIVATE (PRI or PVT) service summarizes virtual storage allocations for private storage in an address space and displays an overview of subpool allocations within a private address space.

Use the PRIVATE service to

- Display allocated storage and fragmented free space by subpool and key
- Debug applications that use excessive private storage
- Tune virtual storage-constrained systems
- Monitor virtual storage use within an address space

Syntax



where

- @ Specifies your own address space; the default.
- asname Is the address space name.
- (asid) Is the address space identifier in decimal format.
- (Xasid) Is the address space identifier in hexadecimal format.
- * Specifies the last address space entered.
- LSQA Displays a detailed report of allocations in the Local System Queue Area, which consists of subpools 205, 215, 225, 236, and 255; the default.
- spi d Is a 3-digit decimal subpool identifier in the range between 0 and 255.
- TCB Displays a summary of the storage allocated by all tasks in the address space.
- address Is the address of the TCB for which you want to display a detailed report of storage allocations.
- HI GH Displays a summary of subpools located in private high storage.

LOW	Displays a summary of subpools located in private low storage.
PVT	Displays a summary of subpools 0–127, 129–132, 229, 230, 236, 237, 251, and 252.
SWA	Displays a detailed report of allocations in the Scheduler Work Area, which consists of subpools 237 and 238.
REGION	Lists regions for the specified address space.

Examples

To determine LSQA usage for address space INVENTORY, type:

private inventory												
AMT	SP	TYPE/KEY	0	1	2	3	4	5	6	7	8-F	TOTAL
AMT	255	ALLOC	240K									240K
AMT		FFS	7K									7K
AMT EXTENDED PRIVATE.....											
AMT	205	ALLOC	4K									4K
AMT		FFS										0K
AMT	215	ALLOC	284K									284K
AMT		FFS	3K									3K
AMT	225	ALLOC	16K									16K
AMT		FFS	8K									8K
AMT	255	ALLOC	8648K									8648K
AMT		FFS	1K									1K
AMT	----- TOTALS -----											
AMT	TOT	ALLOC	9192K									9192K
AMT		FFS	19K									19K

Legend:

1. Type of private storage information displayed.
2. Address space name for which private storage information is displayed.
3. Amount of allocated storage, both above and below the 16MB line, for a subpool within the key.
4. Amount of fragmented free space, both above and below the 16MB line, within the allocated storage for a subpool within the key.
5. Total amount of allocated storage for the address space by subpool and key.
6. Total amount of fragmented free space within allocated storage for the address space by subpool and key.

To summarize storage allocations by subpool and key for all TCBs belonging to address space INVENTORY, type:

```

private inventory tcb

AMTPR8I SUMMARY OF PRIVATE STORAGE FOR INVENTORY BY TCB, SUBPOOL AND KEY
AMTPRBI

AMTPRWI # 1 TCB=7FE158, RB=007FDC58, PGM=IEAVAR00
AMTPRVI =====
AMTPRHI  4  5      ABOVE      BELOW      6
          SP  KEY          ALLOC    FFS     ALLOC    FFS
AMTPRBI  229  0      24K         6K      8K       2K
AMTPRBI  230  0      60K         6K      8K       2K
AMTPRBI -----
AMTPRBI TOTAL          84K        6K      8K       2K 11
AMTPRBI

AMTPRWI # 2 TCB=7FF7C0, RB=007F3D60, PGM=IEFSD060
AMTPRVI =====
AMTPRHI  SP  KEY          ABOVE      BELOW      OWNER
AMTPRHI  ALLOC  FFS     ALLOC    FFS
AMTPRBI  0  0           4K         2K      SHR
AMTPRBI  229  0      4K         3K      36K      6K      OWN
AMTPRBI  230  0     468K      21K      56K     15K      OWN
AMTPRBI  236  0     628K      33K     104K     16K      SHR
AMTPRBI  237  0     636K      33K     124K     21K      OWN
AMTPRBI  252  0     636K      33K     128K     24K      OWN
AMTPRBI -----
AMTPRBI TOTAL          2372K    123K    452K     84K
AMTPRBI

AMTPRWI # 3 TCB=7FF228, RB=007FF128, PGM=IKJEFT01
AMTPRVI =====
AMTPRHI  SP  KEY          ABOVE      BELOW      OWNER
AMTPRHI  ALLOC  FFS     ALLOC    FFS
AMTPRBI  0  0           4K         2K      SHR
AMTPRBI  1  0      12K         5K         8K         3K      SHR
AMTPRBI  2  0           4K         3K         4K         3K      OWN
AMTPRBI  78  0     124K         7K      84K        12K      SHR
AMTPRBI  230  0      44K         7K      56K         6K      OWN
AMTPRBI  236  0     204K        19K     104K         7K      SHR
AMTPRBI  251  0     300K        47K     280K        34K      OWN
AMTPRBI  252  0     500K        52K     480K        43K      OWN
AMTPRBI -----
AMTPRBI TOTAL          1184K    137K   1020K    110K
    
```

Legend:

1. Address of TCB.
2. Address of current RB.
3. Name of program running under the current RB.
4. Subpool number.
5. Key.
6. Whether subpool is owned (OWN) or shared (SHR).

7. Amount of storage allocated above the 16MB line.
8. Amount of free space above the 16MB line.
9. Amount of storage allocated below the 16MB line.
10. Amount of free space below the 16MB line.
11. Total of all allocated and free storage for the TCB, in all subpools and keys.

To display storage allocations by subpool and key for the specific TCB at location 7FE158, type:

```

private inventory tcb 7fe158
AMT1PRFI PRIVATE STORAGE FOR TCB AT 7FE158 BY SUBPOOL AND KEY, AS: INVENTORY
AMT1PRBI
AMT2PRKI 3SP=229 4KEY= 0 4OWN
AMT1PRBI -----
AMT5PRMI 6ADDRESS 6LENGTH 7DATA 8
AMT1PRNI 7FFD0000 0001000 E4C2C640 00000000 00000000 00000000 *UBF ..... *
AMT1PRNI 00000000 00000000 10000000 00000000 *..... *
AMT1PRNI 7FFD2000 0001000 E4C2C640 00000000 00000000 00000000 *UBF ..... *
AMT1PRNI 00000000 00000000 10000000 00000000 *..... *
AMT1PRNI 7FFD4000 0001000 E4C2C640 00000000 00000000 00000000 *UBF ..... *
AMT1PRNI 00000000 00000000 10008000 00000000 *..... *
AMT1PRBI
AMT1PRKI SP=229 KEY= 0 OWN
AMT1PRBI -----
AMT5PRMI 6ADDRESS 6LENGTH 7DATA 8
AMT1PRNI 7FFCA000 0001000 D7C2C640 00000000 00000000 00000000 *PBF ..... *
AMT1PRNI 00000000 00000000 10000400 00000000 *..... *
AMT1PRNI 7FFD1000 0001000 D7C2C640 00000000 00000000 00000000 *PBF ..... *
AMT1PRNI 00000000 00000000 10000400 00000000 *..... *
AMT1PRNI 7FFD3000 0001000 D7C2C640 00000000 00000000 00000000 *PBF ..... *
AMT1PRNI 00000000 00000000 10000000 00000000 *..... *
AMT1PRBI
AMT1PRKI SP=230 KEY= 0 OWN
AMT1PRBI -----
AMT5PRMI 6ADDRESS 6LENGTH 7DATA 8
AMT1PRNI 7F6EC000 0001000 E6001000 00000000 03C40000 00000000 *W.....D... *
AMT1PRNI 00000000 00000000 00000000 00000000 *..... *
AMT1PRNI 7F720000 0002000 00000000 00000000 00000000 00000000 *..... *
AMT1PRNI 00000000 00000000 00000000 00000000 *..... *
AMT1PRNI 7F722000 0001000 E6000A00 00000000 02AF0000 C01E0000 *W..... *
AMT1PRNI 00000000 0000E2E8 E2C2E3E2 D6400005 *.....SYSBTSO... *
AMT1PRBI
AMT1PRKI SP=230 KEY= 0 OWN
AMT1PRBI -----
AMT5PRMI 6ADDRESS 6LENGTH 7DATA 8
AMT1PRNI 007EF000 0001000 00000000 00000000 00000000 00000000 *..... *
AMT1PRNI 00000000 00000000 00000000 00000000 *..... *
AMT1PRNI 007F0000 0001000 00000000 00000000 00000000 00000000 *..... *
AMT1PRNI 00000000 00000000 00000000 00000000 *..... *
AMT1PRNI 7F6E9000 0001000 00000000 00000000 00000000 00000000 *..... *
AMT1PRNI 00000000 00000000 00000000 00000000 *..... *
AMT1PRNI 7FFD5000 0002000 E2E3D6D9 00000000 00001000 C7C4C140 *STOR.....GDA *
AMT1PRNI 7FFD5000 7FFD5120 00000EE0 7FFD5000 *..... *
AMT1PRNI 7FFDA000 0002000 D1C5E2F2 40E3D9C5 40C3D7D6 D6D34040 *JES2 TRE CPOOL *
AMT1PRNI 40404040 40404040 0000015A 7FFDA4C0 *.....
    
```

Legend:

1. Address to which the TCB belongs.
2. Subpool number.
3. Key.
4. Whether subpool is owned (OWN) or shared (SHR).

5. Address of storage block allocated by the TCB in the subpool and key.
6. Length of storage block.
7. First 32 bytes of storage block.
8. Character representation of storage block.

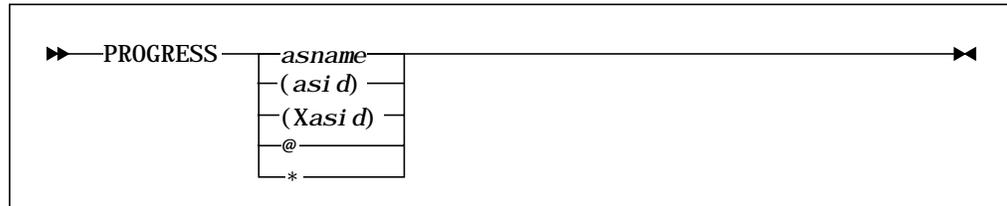
To list the regions for address space INVENTORY, enter:

private inventory region			
AMTPR1I	LIST OF REGIONS FOR PRIVATE ADDRESS SPACE INVENTORY		
AMTPR2I	REGION	ADDRESS	LENGTH
AMTPRBI	RCT	00001000	00004000
AMTPRBI	V=V	00005000	007FB000
AMTPRBI	E V=V	05800000	7A800000

PROGRESS

The PROGRESS (PRO) service displays the status and current step information for a specified address space. It also displays how the number of steps in a job is reached. This service lets you determine how long an address space or step has been running.

Syntax



where

- aname* Is the address space name.
- (asi d)* Is the address space identifier in decimal format.
- (Xasi d)* Is the address space identifier in hexadecimal format.
- @ Specifies your own address space.
- * Specifies the last address space entered.

Example

To display the progress of jobs in address space INVENTORY, type:

```

progress inventory
  AMTP90I JOB 965 INVENTORY IEFPROC PRTY EO (224) PGP 1/1
  AMTP93I MSGCLASS : A MSGLEVEL : (2,0)
  AMTP97I PROGRAM : DFSMVRCO PGMR NAME : **BOOLE**
  AMTP91I JOB START DATE : 90058
  AMTP92I ADDR SPACE START: 17:02:44 JOB CLASS : P
  AMTP94I STEP START : 17:02:44 STEP NO. : 1/ 2

  AMTP95I STEP TCB TIME : 19.36 SRB TIME : 2.23
  AMTP9BI STEP TOTAL CPU : 21.59 CPU LIMIT : 1800.40 1%
  AMTP96I REGION REQUESTED: 2048K EXTENDED : 572K
  AMTP9CI CURRENT USE <16M: 552K EXTENDED : 8K
  AMTP9DI MAX USED <16M: 2.058M EXTENDED : 10.08M
    
```

Legend:

1. JES job ID of the specified address space.
2. Address space name (aname) and current stepname.
3. Address space dispatching priority.

4. Performance group and period.
5. Job message class and allocation and termination message levels.
6. Name of the program that was given control when the current step began execution and the programmer name from the job JCL statement.
7. Time of day the address space was initiated and the initiation job class.
8. Time of day the current step began execution, the current step number, and the total number of steps in the job.
9. Amount of CPU time, in seconds, used by the step in TCB mode, and the amount of CPU time, in seconds, used by the step in SRB mode.
10. Total amount of CPU time, in seconds, used by the current step, the amount of CPU time, in seconds, allotted to the current step, and the percentage of allocated CPU time which has been used.
11. Amount of virtual storage (in 1024-byte increments) requested by the current step and the amount above the 16MB line used by the current step.
12. Amount of virtual storage below the 16MB line currently in use and the amount above the 16MB line currently in use.
13. Highest amount of virtual storage below the 16MB line used by the current step and the highest amount above the 16MB line used by the current step.

PRSM

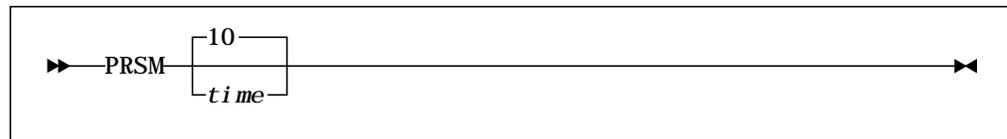
The PRSM service displays CPU busy percentage and processor usage by partition, when running under the IBM PR/SM complex, or by domain, when running in a Multiple Domain Facility (MDF) complex.

When invoked in a PR/SM environment, the PRSM service monitors and reports logical partition activity. The average logical and physical processor utilization is shown for each partition. CPU busy percentage is computed for the current partition.

When invoked in an MDF environment, the PRSM service monitors and reports on individual domain activity. The PRSM service displays minimum, target, maximum, and actual CPU allocations, as well as CPU busy percentage for each domain.

Note: For compatibility with previous releases of MainView for OS/390 (MainView for MVS, and previously called MV MANAGER for MVS), DCPU and PCPU are retained as valid aliases for the PRSM service.

Syntax



where

time Is the length of the sample period in seconds. The default sample period is 10 seconds.

Examples

To display CPU busy percentage and processor usage when running under the IBM PR/SM complex, type:

```

prsm
AMTL01I Collecting statistics for 10 Seconds
AMTLP1I
      ①
AMTLP2I Partitions: 3      Phy. CPUs: 3      Dispatch Intvl: DYNAMIC
AMTLP1I
      ④      ⑤      ⑥      ⑦      ⑧      ⑨      ⑩      ⑪
AMTLP3I
AMTLP4I Partition  Status  Wait  Weight  Relative  Log.  % Log.  % Phy.
AMTLP5I -----  -----  -----  -----  -----  -----  -----  -----
AMTLP6I SYSA      ACT      NO     63     21.0    3     7.9    7.9
AMTLP6I SYSB      ACT      NO    171    57.0    3    58.6   58.6
AMTLP6I SYSC      ACT      NO     66    22.0    3    31.5   31.5
AMTLP1I
AMTLP1I
      ⑫
AMTLPBI Plex busy:      98.1%
      ⑬
AMTLPCI Plex overhead: 1.1%
AMTLP1I
AMTLP1I
      ⑭      ⑮      ⑯
AMTLP7I Current  CPU
AMTLP8I Partition ID  % Busy
AMTLP9I -----  -----
AMTLPAI SYSC      0     47.0
AMTLPAI      1     47.5
AMTLPAI      2     0.0
AMTLP1I

```

Legend:

1. Number of partitions that have been defined on this PR/SM complex.
2. Number of physical CPUs in the PR/SM complex.
3. Number of milliseconds that a virtual processor will get control during a dispatch interval.
4. Name assigned to the logical partition.
5. Status of the logical partition. ACT means that the partition is active. INACT means that the partition has been defined but is not being used.
6. YES indicates that wait assist is available. NO indicates wait assist is not available.
7. Weight that the dispatcher for PR/SM will give to this logical partition.
8. Partition's weight divided by the sum of the weights for all partitions in the plex.
9. Number of logical CPUs that will run in this logical partition.
10. Average logical CPU use that this logical partition is making of the processor complex.
11. Average physical CPU use that this logical partition is making of the processor complex.

- 12. Percentage of time that partitions were dispatched.
- 13. Percentage of time used for dispatching partitions that could otherwise be attributed to a partition.
- 14. Name of the logical partition on which the service is running.
- 15. IDs of the logical CPUs in the current partition.
- 16. Percentage of time each processor in the current partition was busy.

To display information on individual domain activity, when running under an MDF environment, type:

```

prsm

AMTDP7I Statistics being gathered for MDF data
AMTDP1I
AMTDP2I Current Domain: 1PRODMVS      Number: 22
AMTDP1I
AMTDP3I 3Domain      4Domain      5Min      6Tgt      7Max      8CPU      9Int      10Sched
AMTDP4I Name          Number     Alloc     alloc     alloc     Norm     Util     Parm
AMTDP5I -----
AMTDP6I PRODVM        1          10        35        45        37.5    29.8    2
AMTDP6I PRODMVS      2          10        35        45        44.7    43.2    2
AMTDP6I TESTVM       3          1         5         10        4.6     3.7     5
AMTDP6I TESTMWS     4          1         5         10        1.2     .8      5
AMTDP1I
    
```

Legend:

- 1. Name assigned to the current domain.
- 2. Number of the current domain.
- 3. Name assigned to each domain.
- 4. Number of each domain.
- 5. Value set by the operator for the minimum CPU allocation for the domain.
- 6. Value set by the operator for the target CPU allocation for the domain.
- 7. Value set by the operator for the maximum CPU allocation for the domain.
- 8. Actual CPU allocation for the domain.

9. Internal Utilization field. The internal CPU utilization provided by the macrocode is based upon the status (busy or wait) of the processor when the allotted time of a domain ends. It is not a measure of the status during the allotted time.

Because of this, the internal CPU utilization information display

- More precisely represents utilization *relative to* the other domains active during the period.
- Is most precise when the CPUs are closer to 100% busy and least accurate (understated) when the CPUs are only 50% busy.
- Is more precise on 5990 model CPUs where the allotted times are shorter (and more frequent) than on 58xx model CPUs.

On Amdahl 580 or 5890 model CPUs, the source for the Internal Utilization field is not provided or updated by the macrocode, unless

- The FE service key is turned on.
- An Amdahl macrocode patch is applied (see your Amdahl service representative).

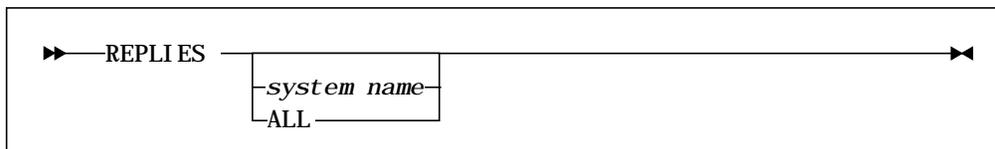
If the FE service is not on or the macrocode patch is not applied, N/A is displayed in the Internal Utilization field.

10. Scheduling parameter for the domains, supplied by the operator on the service console.

REPLIES

The REPLIES (REP) service displays outstanding OS/390 replies with job number, reply number, and the first 56 characters of the message text. Use the REPLIES service to ensure that replies are not overlooked.

Syntax



where

system name Is the 1- to 8-character system name.

ALL Displays replies from all systems.

Note: If you type REPLIES without parameters, only messages from the current system are displayed.

Example

To display outstanding OS/390 replies from the current system, type:

```

replies
AMTR16I  SYSNAME  JOB ID  MESSAGE TEXT
AMTR12I  1SYSB    2STC03223 3*94 DSI802A CNM01 4REPLY WITH VALID NCCF
AMTR17I
AMTR12I  SYSB    STC01830 *87 (IMFSSK) DBC839I REPLY NULL FOR
AMTR17I
AMTR12I  SYSB    JOB01823 *60 DFS996I *IMS READY* X15H
AMTR12I  SYSB    STC00234 @03 ADM2000 I ADMOPUT(ADMPRINT). TO
AMTR17I
                TERMINATE, REPLY 'STOP', 'STOPQ', OR 'STOPS'
  
```

Legend:

1. Name or ID of the system where the replies were issued.
2. JES job ID, including the job number and type (JOB for batch job, STC for started task, and TSU for time-sharing user).
3. Reply number.
4. Reply text.

RESERVES

The RESERVES (RES) service

- Helps determine the cause of poor DASD performance or system lockouts.
- Displays reserve activity for DASD devices.
- Displays system contention caused by long-term reserves (a reserve issued by one processor for a single resource prevents all other processors from accessing the entire volume).

Syntax

```
▶▶——RESERVE——▶▶
```

Example

To display reserve activity on DASD devices, type:

```
reserves
  ①          ②          ③          ④
AMTQ1PI SYSTEM (LOCAL) Q=SYSZVVDS R=CATALOG. ICFMCAT. SYSC
AMTQ1QI SYSID JOBNAME  ASI D  STAT TYP TIME 14: 12: 19
AMTQ1RI SYSB QA7A      (0087) OWNS EXC
AMTQ1RI SYSA FLN1      (0096) WAIT EXC
AMTQ1RI SYSB QA7      (0352) WAIT EXC
AMTQ1LI
AMTQ1PI SYSTEM (LOCAL) Q=SYSGGV2 R=ICFUCAT. VTSG304
AMTQ1QI SYSID JOBNAME  ASI D  STAT TYP TIME 14: 12: 19
  ⑤          ⑥          ⑦          ⑧          ⑨          ⑩          ⑪          ⑫          ⑬          ⑭
AMTQ1RI SYSC MPP1      (0121) OWNS SHR RES=003 PEND 291 TSG304 NR
AMTQ1RI SYSC MPP1      (0121) OWNS SHR CVT
AMTQ1RI SYSB ARG1      (0162) WAIT EXC
```

Legend:

1. Scope of enqueue. The possible scopes are as follows:
 - SYSTEMS
 - SYSTEM
 - STEP
2. Whether the resource is global or local.
3. Major name (Q=).
4. Minor name (R=).
5. System ID for the system executing the task that is holding or waiting for the resource.
6. Jobname, TSO user ID, or started-task ID for the address space containing the task that is holding or waiting for the resource. The jobname is not available in some situations.

7. ID for the address space holding or waiting for the resource.
8. Current status (**OWNS** or **WAIT**), indicating that the task holds (**OWNS**) the resource or is waiting (**WAIT**) for the resource.
9. Type of enqueue: **EXC** for exclusive, **SHR** for shared.
10. **RES** indicates that a reserve is associated with the enqueue. The reserve count is also displayed unless it is zero, in which case the equal sign is also omitted.

CVT indicates that reserve request has been converted to global enqueue. The remaining four fields may not appear.

11. If present, indicates that the reserve is pending, which means that the task is waiting to reserve the devices. Generally, the device is reserved by another system.
12. Device address in hexadecimal format.
13. Volume serial number.
14. If present, indicates that the device is not ready.

RSM

The RSM service provides information on the allocation of real storage within the system. This service also determines the number of pageable and fixed real-storage frames allocated to system common storage areas (LPA, CSA), the system queue area (SQA), and private storage. You can also display the number of frames allocated to each address space.

You can use this information to observe:

- The effects of real-storage page fixing
- The usage of real-storage frames residing above the 16MB line
- The impact of specific workloads on the real-storage resource

Syntax



where

MAP	Provides a tabular display of the amount of real storage allocated to each address space.
ASID	Sorts the output in ascending order by address space identifier.
NAME	Sorts the output in alphabetical order by address space name or system area.
FRAMES	Sorts the output in descending order by the total number of frames allocated to the address space or system area. Only the address spaces with the 10 highest numbers of allocated frames are displayed.
FIXED	Sorts the output in descending order by the total number of frames allocated to the address space or system area that are marked as fixed. Only the address spaces with the 10 highest numbers of allocated fixed frames are displayed.
<16MB	Sorts the output in descending order by the number of frames allocated to the address space or system area that are currently marked as fixed below the 16MB line. Only the address spaces with the 10 highest number of fixed frames below the 16MB line are displayed.

- LSQA Sorts the output in descending order by the total number of frames allocated to the local system queue area for the listed address space. Only the address spaces with the 10 highest number of frames allocated to the local system queue area are displayed.

- WSS Sorts the output in descending order by the working set size of the address space at the time of the last swap-out. Only the address spaces with the 10 largest working set sizes are displayed.

- PERCENT Sorts the output in descending order by the percentage of online real storage frames allocated to the address space. Only the address spaces with the 10 highest percentages of online real storage frame allocations are displayed.

Examples

To display real-storage allocation, type:

rsm		
AMTRS0I	TOTAL	FIXED
	①	②
AMTRS1I PRIVATE FRAMES:	13975	4589
	③	④
AMTRS2I COMMON FRAMES :	2010	289
	⑤	⑥
AMTRS3I SQA FRAMES :	1201	1201
	⑦	⑧
AMTRS4I FREE FRAMES :	495	0
	⑨	⑩
AMTRS5I TOTAL ONLINE :	17681	6079 (>16MB 1175)
	⑪	⑫
AMTRS6I NUCLEUS :	747	747
	⑬	⑭
AMTRS7I HSA FRAMES :	0	0
	⑮	
AMTRS8I *UNKNOWN* :	4	
	⑯	⑰
AMTRS9I TOTAL :	18432 (72M)	6826

Legend:

1. Total frames allocated to private areas.
2. Fixed frames allocated to private areas.
3. Total RCECOMAL-SQA frames allocated to common areas (LPA+CSA).
4. Fixed frames allocated to common areas (CSA+LPA).
5. Frames allocated to SQA (source: OS/390 RSM).
6. Always same as SQA total.
7. Total free frames (calculation: RCEAFC).
8. Always zero.

9. Number of frames currently online (calculation: RCEPOOL is equal to the total of the fields in the column).
10. Total fixed frames allocated in system (calculation: RCETOTFX is equal to the total of the fields in the column).
11. Frames allocated to system nucleus.
12. Same as nucleus total.
13. Frames allocated to hardware storage area (HSA).
14. Always zero.
15. Frames unaccounted for (bad, offline).
16. Total storage on the machine.
17. Total number of fixed frames.

To display the amount of real storage allocated to each address space, type:

```

rsm map
AMTRS0I                TOTAL                FIXED
AMTRS1I PRIVATE FRAMES: 13975                4589
AMTRS2I COMMON FRAMES :   2010                289
AMTRS3I SQA FRAMES   :   1201                1201
AMTRS4I FREE FRAMES  :    495                 0
AMTRS5I TOTAL ONLINE 17681                6079
AMTRS6I NUCLEUS      :    747                747
AMTRS8I *UNKNOWN*    :         4
AMTRS9I TOTAL        :  18432 (72M)        6826

      ①      ②      ③      ④      ⑤      ⑥      ⑦      ⑧
AMTR25I  NAME  ASI D  FRAMES  FIXED  <16MB  LSQA  WSS  PERCENT
AMTR26I  NPDA  0009   113    27    14    24  108    1
AMTR26I  TSOXA01 0013    60    14    14    13   59    1
AMTR26I  IMSMSG 000E    27    15     2    15   47   --
AMTR26I  IMSCTL 000D   300    37    26    29   41    5
AMTR26I  JOBXA01 001F    57    13     3    13   37    1
AMTR26I  VTAM   000B   333    24     8    23   28    5
AMTR26I  DUMPSRV 0005   ----    14   ----    14   25   --
AMTR26I  TCAS   000A   ----    14   ----    14   25   --
AMTR26I  JES3   0008    55    15     2    13   20   --
AMTR26I  ALLOCAS 0007    45    11   ----    11   12   --
AMTR26I  CONSOLE 0006    28    12   ----    12   12   --
AMTR26I  GRS    0004    72    21   ----    17   12    1
AMTR26I  JES3AUX 000C    26    17     5    12   12   --
AMTR26I  PCAUTH 0002    18    15   ----    15   12   --
AMTR26I  TRACE  0003    28    25   ----    25   12   --
AMTR26I  * FREE * **** 1309   ----   ----   ----   ----  23
AMTR26I  *MASTER* 0001    71    44   ----    38   ----    1
AMTR26I  CSA     ****   633   120   178   ----   ----   11
AMTR26I  LPA     ****   643    58   ----   ----   ----   11
AMTR26I  NUCLEUS ****   374   374   ----   ----   ----    6
AMTR26I  SQA     ****   359   359   ----   ----   ----    6
    
```

Legend:

1. The name of the address space or system area whose frame counts are being provided. The first entry (flagged as *FREE*) contains a count of the number of unallocated frames in the system.

The system queue area (SQA), link-pack area (LPA), and common storage area (CSA) table entries follow the FREE entry. The system area entries are followed by the entries for the system address spaces (*MASTER*, PCAUTH, and so on) that are established during system initialization.

Entries for each additional address space in the system are then listed (in ASID order). The last table entry describes the system nucleus.

2. The hexadecimal address space identifier. This field contains asterisks (****) for the free space and system table entries.
3. The total number of frames allocated to the address space or system area.
4. The total number of frames allocated to the address space or system area that are currently marked as fixed. This is a subset of the total frame count provided in the FRAMES field.
5. The total number of frames allocated to the address space or system area that are currently marked as fixed below the 16MB line. This is a subset of the total fixed-frame count provided in the FIXED field.
6. The total number of frames allocated to the local system queue area for the listed address space. This is a subset of the total fixed-frame count provided in the FIXED field.
7. The working set size of this address space at the time of the last swap-out. This field is valid only for swappable address spaces. The nonsystem address spaces, which start out as swappable but later change status to nonswappable, continue to display what appears to be a fixed working set size. In this case, the count actually shows the working set size at the time the address space became nonswappable.
8. The percentage of online real-storage frames allocated to this address space. This is calculated as:

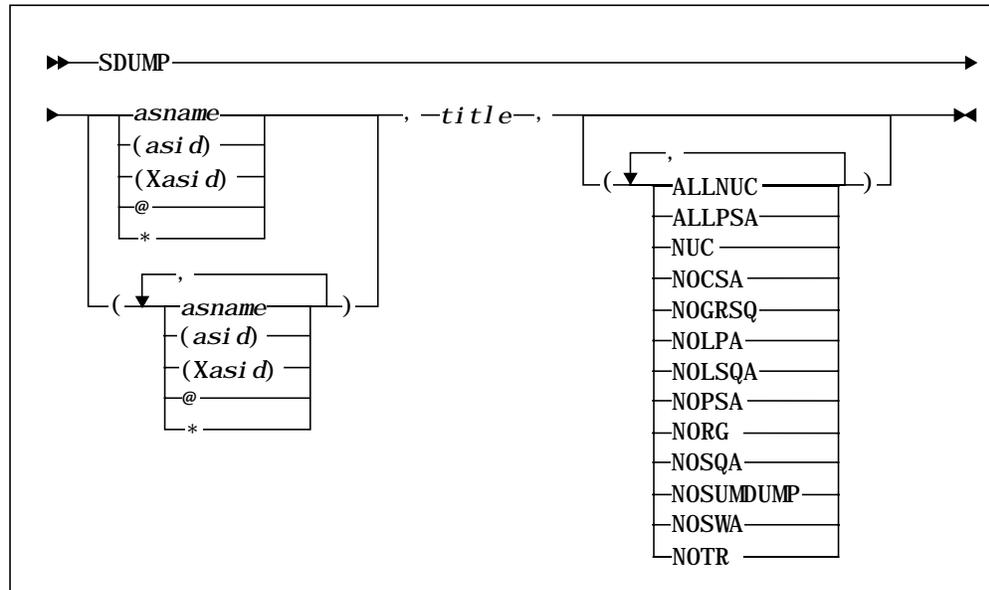
Total frames (pageable and fixed) allocated to the address space or system area,
divided by RCEPOOL, multiplied by 100%.

If the result of this calculation yields a result less than 1%, the field is replaced by dashes (--).

SDUMP

The SDUMP (SD) service takes an SVC dump of one or more address spaces. The SDUMP service works in conjunction with the SYSDUMP service (which lists the SVC dump data sets) and the CLEAR service (which clears the SVC dump data sets). See “SYSDUMP” on page 220 and “CLEAR” on page 61 for more information.

Syntax



where

<i>asname</i>	Is the name of the address space to be dumped. If you specify multiple address spaces, enclose them in parentheses and separate them with commas.
<i>(asi d)</i>	Is the address space identifier in decimal format.
<i>(Xasi d)</i>	Is the address space identifier in hexadecimal format.
@	Specifies your own address space.
*	Specifies the last address space entered.
<i>t i t l e</i>	Is the title to be placed on the SVC dump data set.
ALLNUC	Includes the DAT-ON and DAT-OFF nuclei in the display. These areas are not dumped by default.
ALLPSA	Includes all of the prefixed storage areas (PSAs) in the system. These areas are not dumped by default.
NUC	Includes the non-page-protected areas of the DAT-ON nucleus in the display. These areas are not dumped by default.

NOCSA	Suppresses display of the CSA and ECSA subpools. These areas are dumped by default.
NOGRSQ	Suppresses display of the global resource serialization control blocks. These control blocks are dumped by default.
NOLPA	Suppresses display of the active link-pack area modules and SVCs. These areas are dumped by default.
NOLSQA	Suppresses display of the LSQA and ELSQA subpools. These subpools are dumped by default.
NOPSA	Suppresses display of the PSA for one processor. This is either the processor at the time of the error or the processor at the time of the dump. The PSA is dumped by default.
NORGN	Suppresses display of the allocated pages in the address space's private area. These pages are dumped by default.
NOSQA	Suppresses display of the SQA and ESQA subpools. These subpools are dumped by default.
NOSUMDUMP	Suppresses display of a summary dump. This dump is included by default.
NOSWA	Suppresses display of the scheduler work area subpools for the address space. These subpools are dumped by default.
NOTRT	Suppresses display of the system trace table, the GTF trace records, and master track data, if these traces are active. These areas are dumped by default.

Note: These areas are defined by IBM. For more information, see the IBM publication, *Authorized Assembler Programming Reference*.

Examples

To obtain a dump address space INVENTORY, adding the title "DUMP1," type:

```

sdump inventory dump1

AMTSD7I    DUMP IN PROGRESS           ①
AMTSD2I    COMPLETE DUMP TAKEN FOR INVENTORY ②
```

To obtain a dump of address space INVENTORY, the address space with ASID 280, and your own address space, type:

```
sdump (inventory, (280), @)

AMTSD7I  DUMP IN PROGRESS           ❶
AMTSD7I  DUMP IN PROGRESS
AMTSD7I  DUMP IN PROGRESS
AMTSD2I  COMPLETE DUMP TAKEN FOR INVENTORY  ❷
AMTSD2I  COMPLETE DUMP TAKEN FOR ASID 280
AMTSD2I  COMPLETE DUMP TAKEN FOR @
```

To obtain a dump of address space INVENTORY, adding the title “DUMP2” and suppressing the trace table dump and summary dump, type:

```
sdump inventory, dump2, (notrt, nosumdump)

AMTSD7I  DUMP IN PROGRESS           ❶
AMTSD2I  COMPLETE DUMP TAKEN FOR INVENTORY  ❷
```

Legend:

1. Indicates that an SVC dump is being taken.
2. Indicates that an SVC dump has been successfully created.

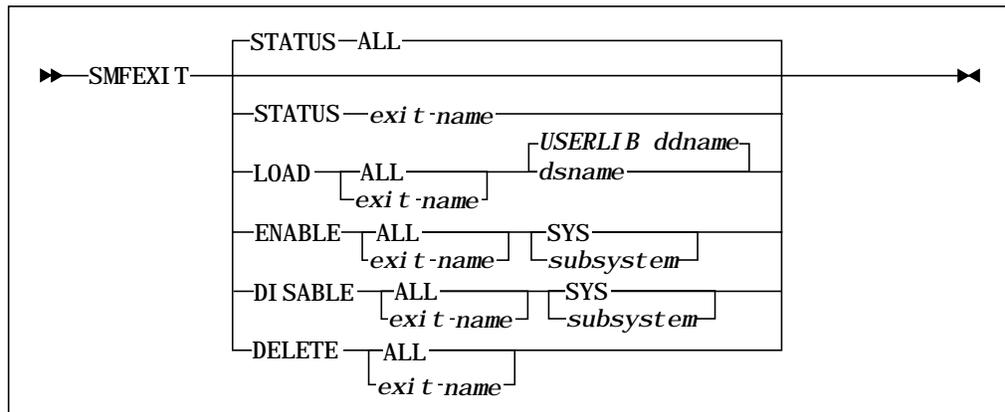
SMFEXIT

The SMFEXIT services is documented for users running versions of MVS below MVS SP 5.1 and MVS SP 5.1 and above.

Below to MVS SP 5.1

The SMFEXIT (SMF) service loads, deletes, enables, disables, and displays the status of SMF exits. This service also lets you test new SMF exits without having to IPL the system.

Syntax



where

STATUS

Displays the status of all SMF exits previously loaded by the SMFEXIT service on behalf of the user since the last IPL.

LOAD

Scans the specified data set for one or all SMF exit load module names. If a data set name is not specified, SMFEXIT searches the load library defined by the DD name USERLIB. An attempt is made to load each selected exit into global CSA storage. Exit load modules marked RMODE ANY are loaded into extended CSA space. All other exits are loaded into CSA space below the 16MB line. A record is made of each exit successfully loaded. Any exit loaded is therefore visible to subsequent invocations of SYSPROG Services. The loading of an SMF exit routine does not affect the operation or status of any SMF exit currently in use.

ENABLE Activates one or all previously loaded SMF exits. Valid subsystem names are SYS, STC, TSO, and JES x , where x is the name of the JES subsystem in use (JES2 or JES3).

To enable an exit:

1. Locate the appropriate SMF subsystem table.
2. Locate the appropriate exit control table for the subsystem.
3. Save the status and address of the currently defined SMF exit.
4. Update the exit control table to define the address of the new SMF exit (previously loaded by the SMFEXIT LOAD function).
5. Mark the new exit as active in the SMF exit control table.

Should any step in this operation fail, an appropriate error message is issued and the status of the current SMF exit is not altered.

DI SABLE Deactivates one or all previously enabled SMF exits. The process of disabling a given SMF exit is the reverse of enabling an exit. The result is to return the status of the exit to the status that existed prior to the disabled exit becoming current.

DELETE Deletes one or all previously loaded SMF exits. Enabled exits may not be deleted.

To delete an exit:

1. Locate the global SYSPROG Services record of the exit.
2. Ensure that the exit is not enabled for processing by any SMF subsystem.
3. Free the CSA storage in which the exit load module exists.
4. Update the global records to delete all references to the deleted exit.

ALL Specifies all exit name operands (see the *exit-name* operand).

exit-name Specifies the exit name on all service requests. It can be one of the following:

Operand	Description
ACTRT	IEFACTRT step/job end exit
UJI	IEFUJI job/TSO session initiation exit
UJP	IEFUJP job purge exit
UJV	IEFUJV job validation exit
USI	IEFUSI step initiation exit
UTL	IEFUTL time limit exit
USO	IEFUSO sysout limit exit
U29	SMF dump exit
U83	IEFU83 SMF record-writer SVC exit
U84	The IEFU84 SMF record-writer branch exit

USERLIB ddname Is a DD name. If no data set name is specified, SMFEXIT tries to find the modname in the DD concatenation USERLIB. USERLIB must have been previously allocated to the same address space where SYSPROG Services is currently executing.

<i>dsname</i>	Is the name of the data set containing the SMF exit to be loaded.								
<i>SYS</i>	Indicates the global system level subsystem.								
<i>subsystem</i>	Specifies one of the following subsystem names: <table> <tr> <td><i>SYS</i></td> <td>Global system level subsystem used when the equivalent exit or option is not specified at one of the other subsystem levels.</td> </tr> <tr> <td><i>STC</i></td> <td>Started-task subsystem exits and options.</td> </tr> <tr> <td><i>TSO</i></td> <td>Time-sharing user-subsystem exits and options.</td> </tr> <tr> <td><i>JESx</i></td> <td>Batch job subsystem exits and options, where <i>x</i> is the identification number (2 or 3) for the JES subsystem in use.</td> </tr> </table>	<i>SYS</i>	Global system level subsystem used when the equivalent exit or option is not specified at one of the other subsystem levels.	<i>STC</i>	Started-task subsystem exits and options.	<i>TSO</i>	Time-sharing user-subsystem exits and options.	<i>JESx</i>	Batch job subsystem exits and options, where <i>x</i> is the identification number (2 or 3) for the JES subsystem in use.
<i>SYS</i>	Global system level subsystem used when the equivalent exit or option is not specified at one of the other subsystem levels.								
<i>STC</i>	Started-task subsystem exits and options.								
<i>TSO</i>	Time-sharing user-subsystem exits and options.								
<i>JESx</i>	Batch job subsystem exits and options, where <i>x</i> is the identification number (2 or 3) for the JES subsystem in use.								

Note: You can abbreviate the first operand to a unique character. For example, you can abbreviate *STATUS* to *S*; however, you cannot abbreviate *DI SABLE* or *DELETE* to *D* because of the conflict; instead, abbreviate *DI SABLE* to *DI* and *DELETE* to *DE*.

Examples

The following examples illustrate the ways to invoke the SMFEXIT service.

STATUS Example

To display the status of the previously loaded SMF exits, type:

```

smfexit
  ①      ②      ③      ④
AMTS40I IEFUJI : XADDR=10BA41A0 EP=109C1020 STATUS=ACTIVE

```

Legend:

1. Name of an SMF exit previously loaded by the SMFEXIT service.
2. Address of the SMF exit previously loaded by the SMFEXIT service.
3. Entry point address of the exit.
4. Status of the exit. Valid status codes are as follows:

INACTIVE Exit is loaded but has not been enabled.

ACTIVE Exit is both loaded and enabled for execution by the SMFEXIT service.

The following additional status line is produced only if the exit is in *ACTIVE* status, as indicated in the *STATUS* field:

```

      ⑤      ⑥
AMTS41I      : RADDR=10BA41A0 SUBSYS=SYS

```

5. Address of the SMF exit that was active prior to enabling the current exit, whose address is in the *XADDR* field. This exit becomes current if and when an SMFEXIT service request to disable the SMF exit named in field 1 is received.

- Name of the subsystem for which the exit is enabled.

LOAD Example

To load all SMF exits present in the library defined by the USERLIB file definition, type:

```
smfexit load all
AMTS42I SMF EXIT 'IEFACTRT' SUCCESSFULLY LOADED
```

Legend:

- Name of an SMF exit loaded from the library defined by the USERLIB file or the ddname defined on the service request. The USERLIB or ddname-defined library does not have to be APF-authorized.

If the named SMF exit load module is not marked re-entrant (that is, RENT), the following warning message is issued:

```
AMTS41W SMF EXIT MODULE 'IEFACTRT' IS NOT MARKED RE-ENTRANT
```

Loading of the module is completed on the assumption that the load module is re-entrant. If the exit module is not re-entrant, serious system damage may occur if the named exit is enabled subsequently for processing by SMF.

ENABLE Example

To enable a test version of the IEFUJI SMF exit previously loaded by the SMFEXIT service, type:

```
smfexit enable uji
AMTS44I SMF EXIT 'IEFACTRT' ENABLED ON 94.022 AT 14:22:02 FOR 'SYS'
```

Legend:

- Name of an SMF exit previously loaded by the SMFEXIT service and enabled for SMF processing. A restorable copy of the exit is saved for use by the DISABLE function unless the exit was previously undefined to SMF.

DISABLE Example

To disable a test version of the IEFACRT SMF exit previously enabled by the SMFEXIT service, type:

```
smfexit disable actrt
AMTS45I SMF EXIT 'IEFACTRT' DISABLED
```

Legend:

1. Name of an SMF exit loaded and enabled previously by the SMFEXIT service and disabled for SMF processing. The restorable copy of the exit is made the active copy unless the exit was previously undefined to SMF, in which case the exit is deactivated.

Note: The DISABLE function requests that one or all previously enabled SMF exits be deactivated. The process of disabling a given SMF exit is the reverse of enabling an exit. The status of the exit returns to what it was before the disabled exit became current.

DELETE Example

To release the storage for all test SMF-exit modules previously loaded by the SMFEXIT service, type:

```
smfexit delete all
AMTS43I SMF EXIT 'IEFACTRT' DELETED
```

Legend:

1. Name of an SMF exit previously loaded by the SMFEXIT service and deleted. The named exit must be in INACTIVE status for the DELETE function to be successful. The CSA storage occupied by the exit module is released.

Note: The DELETE function requests that one or all previously loaded SMF exits be deleted from the system. Enabled exits cannot be deleted.

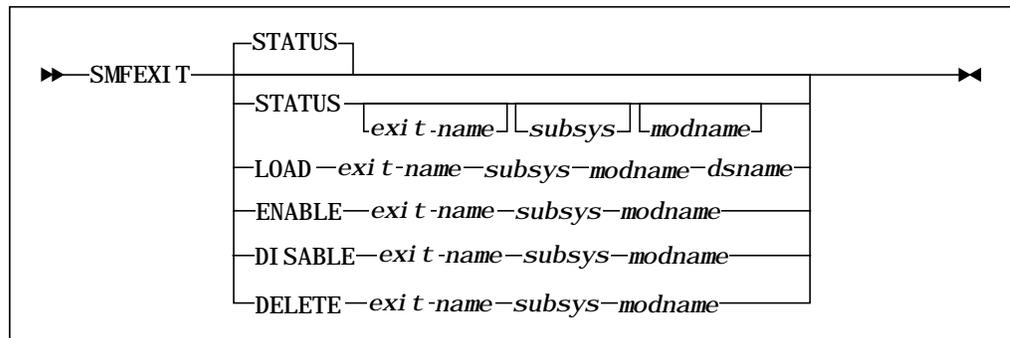
MVS SP 5.1 and Above

The SMFEXIT (SMF) service loads, enables, disables, and deletes SMF exit routines and displays the status of SMF exits. Use this service to dynamically add or delete SMF exit routines.

An **exit point** (also called simply an exit) calls the exit routine. The exit point is uniquely identified by a combination of the SMF exit name and the subsystem name. Multiple exit routines may be loaded and enabled for a given exit point.

An **exit routine** is a load module that is called by an exit point. The exit point can establish requirements for the exit routine. The exit can require that the exit routine be re-entrant or that it be in a certain addressing mode. All exit routines must be loaded from an APF-authorized data set. The same exit routine may be used with multiple exit points if designed for this purpose. It is recommended that user-written or third-party exit routines be given names that do not duplicate existing IBM exit names. For example, use XYZUJV instead of IEFUJV.

Syntax



where

STATUS Displays the status of all or selected SMF exits. All exits are displayed unless parameters are specified to limit the display.

For example, if you specify the exit name, all exit routines for the specified exit name will be displayed. If you specify a subsystem name, all exit routines for the subsystem will be displayed. If you specify both an exit name and a subsystem name, only the exit routines for that subsystem and exit will be displayed. If you specify a module name, all exits using that module name will be displayed.

LOAD Causes the load module (exit routine) to be associated with the specified SMF exit point. The exit point is identified by the SMF exit name and the subsystem name.

Notes: The operating system does not actually load the module until you **ENABLE** the exit. However, validation is performed by the SMFEXIT **LOAD** operation. If successful, the exit routine you have loaded will be loaded as **DISABLEd**. To activate the exit, you must use the **ENABLE** function.

Multiple exit routines may be associated with the same exit as long as they have different load module names. To reload the exit routine (**MODNAME**) for the same exit, you must first **DISABLE** and then **DELETE** the exit routine. New versions of an exit routine should be uniquely named so that the prior version can be retained (in a **DISABLEd** status). This way, it can be reactivated (**ENABLEd**) if the new exit routine is in error.

You must specify the exit name, subsystem name, and load module name on the **LOAD** request. The data set name is also required unless previously specified.

ENABLE Causes the specified exit routine (**MODNAME**) to be **ENABLEd** (activated) for the specified exit point.

When initially **LOADed**, modules are **DISABLEd** and therefore cannot be called by the exit point until they are **ENABLEd**.

The SMF exit name and the subsystem name are combined to form an exit identifier that uniquely identifies the exit point. One or more exit routines may be associated with (called by) a single exit point (or exit). The exit routine is identified by its load module name (**MODNAME**).

DI SABLE Causes the exit routine for the specified exit to be **DISABLEd**. The exit routine remains in storage and may be **ENABLEd** again unless the **DELETE** function is used to **DELETE** the exit routine.

You must specify the exit name, subsystem name, and load module name to **DISABLE** the exit routine.

DELETE Causes the exit routine to be removed from storage. The exit routine must be **DISABLEd** before it can be deleted.

You must specify the exit name, subsystem name, and load module name to **DELETE** the exit routine.

exit-name Name of the SMF exit that is to be **LOADed**, **ENABLEd**, **DISABLEd**, **DELETEd**, or displayed by the SMFEXIT service. Valid SMF exit names are **ACTRT**, **UJI**, **UJP**, **UJV**, **USI**, **USO**, **UTL**, **U29**, **U83**, and **U84**.

	<p>The SMF exit name and the subsystem name are combined to form an exit identifier that uniquely identifies the exit point. One or more exit routines may be associated with (called by) a single exit point (or exit). The exit routine is identified by its load module name (MODNAME).</p>
<i>subsys</i>	<p>Name of the subsystem. The subsystem name to be used to identify the exit point. For STATUS requests, the subsystem name can be specified individually or in combination with other parameters to limit the display.</p> <p>Valid subsystem names are: SYS, TSO, JES2, and JES3.</p> <p>The SMF exit name and the subsystem name are combined to form an exit identifier that uniquely identifies the exit point. One or more exit routines may be associated with (called by) a single exit point (or exit).</p>
<i>modname</i>	<p>Load module name for the exit routine. An exit routine may be associated with (loaded and enabled for) multiple exit points if desired. An exit point is uniquely identified by an exit identifier which is a combination of the SMF exit name and the subsystem name.</p> <p>In addition, multiple exit routines (each with a unique load module name) may be loaded and enabled for a given exit point.</p> <p>However, multiple versions of an exit routine with the same load module name are not supported.</p> <p>The default module name is the IBM exit routine name. However, it is recommended that you use a name other than the IBM name when creating an exit routine.</p>
<i>dsname</i>	<p>Name of the data set containing the exit routine (MODNAME) that is to be LOADED. When specified, the data set name is retained for the life of the MainView for OS/390 session and therefore need not be respecified on subsequent LOAD requests.</p> <p>The data set must be a partitioned data set. It must be cataloged and it must be APF authorized. If it is not APF authorized, the MainView for OS/390 APF service may be used to authorize it.</p>

Usage Notes

- Exit routines are DISABLED when LOADED. The exit routine will not be called by the SMF exit until it has been ENABLED.
- The exit routine is not actually loaded into storage until it is ENABLED. However, LOAD processing validates the exit routine.
- For LOAD requests, the data set name is required. However, the data set name is retained for the life of the MainView for OS/390 session and therefore need not be re-entered on subsequent load requests.
- The SMF exit (exit) calls the exit routine. You can LOAD and ENABLE multiple exit routines for the same exit. You can also LOAD and ENABLE the same exit routine for multiple SMF exits.
- Although multiple exit routines may be LOADED and ENABLED for a single exit, each exit routine must have a unique name. For this reason, it is recommended that you do not

give your exit routine the same name as that of a supplied IBM exit routine. For example, use MYUJV instead of IEFUJV.

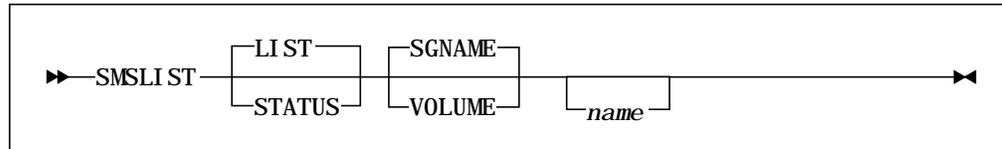
- An SMF exit is uniquely identified by the combination of the SMF exit name and the subsystem name. For example, SMF exit UJV and subsystem STC combine to identify an exit. SMF exit UJV and subsystem SYS combine to identify a different exit.
- For STATUS requests, any combination of parameters may be specified to limit the display. If no parameters are specified, all exit routines will be displayed.

SMSLIST

The SMSLIST (SMS) service applies to OS/390 systems that use system-managed storage (SMS). For DASD volumes under control of SMS, the SMSLIST service displays:

- Storage group names
- Space usage

Syntax



where

- | | |
|---------------|---|
| LI ST | Lists either DASD volumes and storage group names or storage groups and their associated DASD volume names as indicated by the second parameter; the default. |
| STATUS | Lists storage group and volume names for DASD volumes along with their status, space usage, percentage of free space, and the largest free extent. |
| SGNAME | Specifies that information about storage groups is listed; the default. If you specify SGNAME with STATUS , the SMS information is listed in ascending alphanumeric order by storage group. If you specify SGNAME with LI ST , the storage group names and the names of all DASD volumes in the storage group are listed. |
| VOLUME | Specifies that information about DASD volumes is listed. If you specify VOLUME with LI ST , a list of DASD volumes and each volume's storage group name is listed. If you specify VOLUME with STATUS , the SMS information is listed in ascending alphanumeric order by volume name. |
| <i>name</i> | Limits the information displayed to the specified volume or storage group name you enter. You can end <i>name</i> with an asterisk (*) to request a generic search (for example, <i>MVS*</i> for all names starting with <i>MVS</i>).

If you specify SGNAME , you can type any storage group name. If you specify VOLUME , you can type any DASD volume ID. |

Examples

To list DASD volumes and each volume's storage group name, type:

```
smslist list volume
AMTSSB1 VOLUME  GROUP      VOLUME  GROUP
AMTSSB2 -----  -----  -----  -----
AMTSSCI BBSMS1  BBSG
```

To display the status, space usage, percentage of free space, and the largest free extent for storage group BBSG, type:

```
smslist status sname bbsg
AMTSSF1 -- SG -- VOLUME  ---- STATUS  ----- SPACE ----- ERROR
AMTSSF2  NAME   ID      MVS     SMS     FREE  L-XTENT  TOT CAP  FLAG
AMTSSF3                                     (MB)    (MB)
AMTSSF4 -----  -----  -----  -----  -----  -----  -----
AMTSSCI BBSG      BBSMS1  ONLINE  ENABLED  99.2%    300     602
```


dsname Is the name of the data set assumed to be a LOGREC data set when a value other than LOGSTREAM is specified as the fourth parameter. The records are obtained from this data set even if it is not currently being used for recording errors. To display records from the current recording media, omit the fourth parameter.

Note: If you type SOFTFRR without operands, the service summarizes all software records in the LOGREC data set.

Examples

To summarize all software records, type:

```

soft
AMTL3NI Processing logstream SYSPLEX.LOGREC.ALLRECS
AMTL31I 67 Records read; 17 Records accepted
AMTL39I N/A 00003 IXGINLPA 00003 IGC0013I 00001
AMTL39I IKJEFLC 00003 ISLOAD 00001 NUCLEUS 00001
AMTL39I BBM33 00001 XDCCALL 00001 IMF 00001
AMTL39I BBI 00002
          ①          ②

```

Legend:

1. Lists all modnames for which there is data. N/A (not available) means that the module identifier consists of binary zeros.
2. Shows the number of records for each modname.

To display all of the data for records containing the modname BBM33, type:

```

soft, bbn33, map

AMTL3NI Processing logstream SYSPLEX.LOGREC.ALLRECS
AMTL3QI Record from system SYSC 1
AMTL32I ErrorID: CPUID: 0000; SeqNo: 296; Time: 17.15.57.8; ASID: 0028
AMTL33I Abend U1900 on 04/16/97 at 17.15.58.09 -- BBM33
BBMOSF20.BBMOZE20

AMTL34I Err-PSW: 070C1000 FF6E4020; ILC 2; INTC 000D; Trans: 00000000 2
AMTL3HI Data at PSW-6 47F0, 9E16, 0A0D, 0003, 0001, 0000
AMTL35I EFLGS: Task issued SVC 13 | Enabled RB in control
AMTL35I : Service information available | Abnd due to ancestor task error
AMTL35I : ErrorID information available |
AMTL36I Err-Regs 0-3 00000001 8400076C FF6E32C8 11BB1880
AMTL36I Err-Regs 4-7 7F6E4784 11BAD7BC 00000008 11BAD91C
AMTL36I Err-Regs 8-11 91BB1956 7F6E3208 7F6DB8D0 007C8C78
AMTL36I Err-Regs 12-15 7F6DF520 7F6DB720 FF6E32C8 00030001
AMTL37I RB-PSW: 070C0000 91BB1CAC; ILC 2; INTC 000D 3
AMTL38I RB-Regs 0-3 00000001 7F4BA8D0 00000040 7F4BA8D0
AMTL38I RB-Regs 4-7 11BAD1D8 11BAD7BC 00000008 11BAD91C
AMTL38I RB-Regs 8-11 91BB1956 91BB1880 7F6DB8D0 007C8C78
AMTL38I RB-Regs 12-15 7F6DF520 7F6DB720 91BB1CAC 7F6E3208
AMTL3EI Recovery routine percolated error with 'No ESTAI/STAI' Option 4
AMTL3GI RFLGS: LOGREC recording requested

AMTL3AI Variable Recording Area (VRA) Data 5

AMTL3II VRAPID Product ID: BBM#ZZ
AMTL3II VRAHID Header ID: SEQN
AMTL3II VRAEHX Error information in hex: 00000001
AMTL3II VRAHID Header ID: NRRC
AMTL3II VRAEHX Error information in hex: 01
AMTL3II VRAHID Header ID: FLGS
AMTL3II VRAHEX Information in hex: 7014044010000000
AMTL3II VRAPA Execution path trace data: BBMOSF20

```

Note: The above record was obtained from the logstream.

Legend:

1. Displays the name of the system where data has been obtained.
2. Displays the error level PSWs and registers. The six bytes of data preceding and following the PSW address at the time of the abend are also shown.
3. Displays the RB level PSW and registers.
4. Translates the status flags.
5. Displays variable recording area (VRA) data, if any.

To display the modname and map for records that have data coming from the LOGREC, type:

```

soft, ikjeft01, map

AMTL30I Software Logrec Report (Vol=ES430M; Dsn=SYS1.LOGREC)

AMTL32I ErrorID: CPUID: 0000; SeqNo: 937; Time: 14.26.53.3; ASID: 008A
AMTL33I Abend S13E on 04/23/97 at 14.26.53.68 -- IKJEFT01.IKJEFTSC.IKJEFT05

AMTL34I Err-PSW: 078C0000 80077A30; ILC 2; INTC 0001; Trans: 00000000 ①
AMTL3HI Data at PSW-6 4100,0001,0A01,D503,401C,C4E8
AMTL35I EFLGS: System issued SVC 13 | Enabled RB in control
AMTL35I :Cleanup only - Retry not allowed|Abnd due to ancestor task error
AMTL35I :ErrorID information available |
AMTL36I Err-Regs 0-3 00000001 11A07F98 00000000 00050A38
AMTL36I Err-Regs 4-7 11A07F7C 11A323E8 00000000 000C86E0
AMTL36I Err-Regs 8-11 00050A38 0010F768 00077B84 0010EF28
AMTL36I Err-Regs 12-15 80077740 000C8330 00077B7C 807B8658
AMTL37I RB-PSW: 071C1000 810FF542; ILC 2; INTC 000D ②
AMTL38I RB-Regs 0-3 00000001 FF84C804 83F75728 03F76727
AMTL38I RB-Regs 4-7 007B7418 00000000 007B3E20 007FE030
AMTL38I RB-Regs 8-11 00000000 807B3D48 007B49B8 03F77726
AMTL38I RB-Regs 12-15 007B32F0 007B32F0 007B3D48 807B76C8
AMTL3DI Recovery routine percolated error ③
AMTL3GI RFLGS: LOGREC recording requested

AMTL3AI Variable Recording Area (VRA) Data ④

AMTL3II VRACBM Control block macro name: TMPWRKA2
AMTL3II VRADAE DAE indicator:
AMTL3II VRACBA Control block address: 00005000F2F8

AMTL31I 839 Records read; 1 Records accepted

```

Note: The above record was obtained from SYS1.LOGREC.

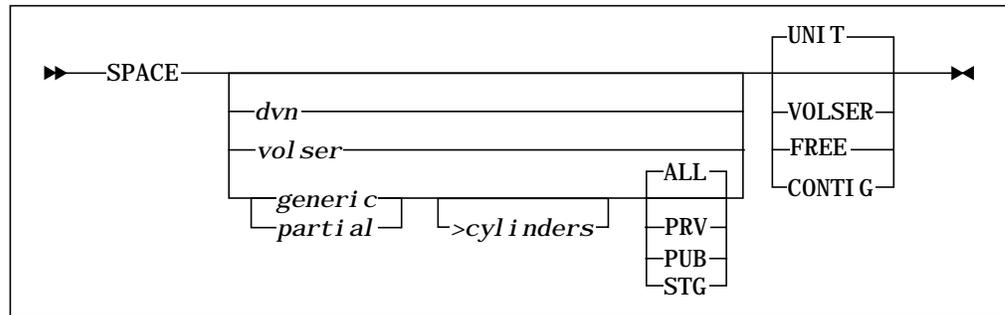
Legend:

1. Displays the error level PSWs and registers. The six bytes of data preceding and following the PSW address at the time of the abend are also shown.
2. Displays the RB level PSW and registers.
3. Translates the status flags.
4. Displays variable recording area (VRA) data, if any.

SPACE

The SPACE (SPA) service determines the amount of space available on a set of specified devices and displays the amount by device number. Use this service to find volumes that contain a specified minimum amount of available contiguous space, in cylinders.

Syntax



where

- dvn* Is a device number.
- volser* Is a complete six-character volume serial number.
- generic* Is a generic or esoteric device name.
- partial* Is the first one to five characters of a volume serial number.
- >cylinders* Specifies a minimum amount of available contiguous space, in cylinders. When this parameter is specified, the output is sorted in descending order by the CONTIG field, unless specified otherwise. When this parameter is omitted, the output is sorted in descending order by the FREE field, unless specified otherwise.
- ALL Displays both online and offline devices.
- PRV Specifies that only devices with a mount attribute of PRV (Private) be displayed.
- PUB Specifies that only devices with a mount attribute of PUB (Public) be displayed.
- STG Specifies that only devices with a mount attribute of STG (Storage) be displayed.
- UNIT Specifies that the output be sorted in ascending order by unit address; the default.
- VOLSER Specifies that the output be sorted in ascending order by volser.
- FREE Specifies that the output be sorted in descending order by the amount of free space. Only the 10 devices or volsers with the highest amount of free space are displayed.

CONTIG Specifies that the output be sorted in descending order by the amount of contiguous free space. Only the 10 devices or volumes with the highest amount of contiguous free space are displayed.

Examples

To display disk space for the SYSALLDA esoteric unit name, type:

```

space sysallda free
      ①          ②          ③
AMTS10I ESOTERIC UNITNAME SYSALLDA DEFINES      3 DEVICES
      ④      ⑤ ⑥          ⑦          ⑧
AMTS11I UNIT B13 SYSP3B PUB FREE= 130 CYL/ 21 TRK; CONTIG= 100 CYL/ 11 TRK
AMTS11I UNIT 500 SYSR1A STR FREE= 124 CYL/ 71 TRK; CONTIG= 13 CYL/ 0 TRK
AMTS11I UNIT 240 SYSP2B PRV FREE= 20 CYL/ 21 TRK; CONTIG= 10 CYL/ 11 TRK

```

Legend:

1. GENERI C or ESOTERI C.
2. Requested generic or esoteric unitname.
3. Number of devices described by the generic or esoteric unitname.
4. Unit address.
5. Volume name.
6. Device type:
 - PRV is PRIVATE.
 - PUB is PUBLIC.
 - STR is STORAGE.
7. Number of free cylinders and tracks.
8. Largest number of contiguous cylinders and tracks.

To display disk space for devices that start with a partial volume serial number of TNT and have more than 20 cylinders of contiguous free space, type:

```

space tnt >20
AMTS11I UNIT B13 TNT001 PRV FREE= 16 CYL/ 260 TRK; CONTIG= 12 CYL/ 10 TRK
AMTS11I UNIT B13 TNT002 PRV FREE= 40 CYL/ 260 TRK; CONTIG= 23 CYL/ 1 TRK
AMTS11I UNIT B13 TNT003 PUB FREE= 83 CYL/ 260 TRK; CONTIG= 44 CYL/ 9 TRK
AMTS11I UNIT B13 TNT004 STR FREE= 109 CYL/ 260 TRK; CONTIG= 100 CYL/ 3 TRK

```


To display the addresses of SRM control blocks as well, type:

```

srm map 1
AMTS30I PARMLIB MEMBERS: IPS=01          OPT=00          ICS=00
2          3          4
AMTS31I ADDRESS SPACES : IN= 45          0/RDY= 0        0/NRDY= 62      LOGWT=nn
5          6          7
AMTS32I RESOURCE STATUS: CPU= 57%        LTCPU= 66%      HIGH UIC= 11
AMTS36I PAGING RATE/SEC: DEMAND= 23      TOTAL= 29       DELAY TIME= 40
AMTS39I THINK TIME(SEC): CURR= 10.0     MIN= 9.0        MAX= 60.0

RMCT 0111B2A0  RMCA 0111B908  RMPT 0111B898  CCT 0111B4C8
ICT 0111B5C8   MCT 0111B630  LSCT 0111BA68  DMDT 01977D90
CMCT 0111BAA8  CPMI 0196EE20  CMB  01EFE000  CPWK 0196EE00

```

Legend:

1. Name of the installation performance specification member, currently in effect, in the SYS1.PARMLIB library.
2. Number of address spaces currently swapped in memory.
3. Number of address spaces swapped out of memory but ready.
4. Number of address spaces out of memory and not ready.
5. Current SRM-weighted percentage of CPU busy.
6. Long-term SRM-weighted percentage of CPU busy.
7. Highest unreferenced interval count.

To display system resource information for address space INVENTORY, type:

```

srm inventory
AMTS71I JOB 1283 INVENTORY STEP1 PRTY 26(111) PGP 32/1
AMTS72I TRANSACTION DATA: INTERVAL 2:26:14 DOMAIN 3
4
AMTS73I LAST SWAP : INTERVAL 0:00:16 REASON IN MEMRY COUNT 254
5
AMTS74I SWAPPING PROFILE: COUNT 1 PGS-OUT N WRK-SET 91
6
AMTS75I SERVICE DATA : CPU 29205 I/O 6450 MSO 4100 TOTAL 39755
7
8

```

Legend:

1. Job ID, jobname, and stepname for the specified job.
2. Priority and internal dispatching priority.
3. Performance group and period.
4. Duration of this particular transaction. The DOMAIN value is the IPS domain in which this job resides.

5. The LAST SWAP INTERVAL is the time since the last swap action. If the job is in memory, this is the time it has been in; if it is out of memory, it is the time it has been swapped out.

The REASON code shows the cause of the current swap status. Allowable codes are as follows:

Code	Description
ASM LOW	Swapped out due to auxiliary storage shortage
ENQ EX	Swapped out due to enqueue exchange
IN MEMRY	Swapped in
LONG WT	Swapped out due to long wait
MSO WAIT	Swapped out due to MSO-detected long wait
REC EX	Swapped out due to recommendation values
REQUEST	Swapped out due to a request swap
RSM LOW	Swapped out due to a real-storage shortage
TERM IN	Swapped out due to terminal input wait
TERM OUT	Swapped out due to terminal output wait
UNI LTRL	Swapped out due to unilateral swap-out

6. Number of swaps during the life of the current transaction.
7. Pages swapped at last swap-out and the working-set size at swap-in.
8. Service units for this transaction. Values are as follows:

Name	Description
CPU	CPU service units
I/O	I/O service units
MSO	Main storage occupancy service units
TOTAL	Total service units for this transaction

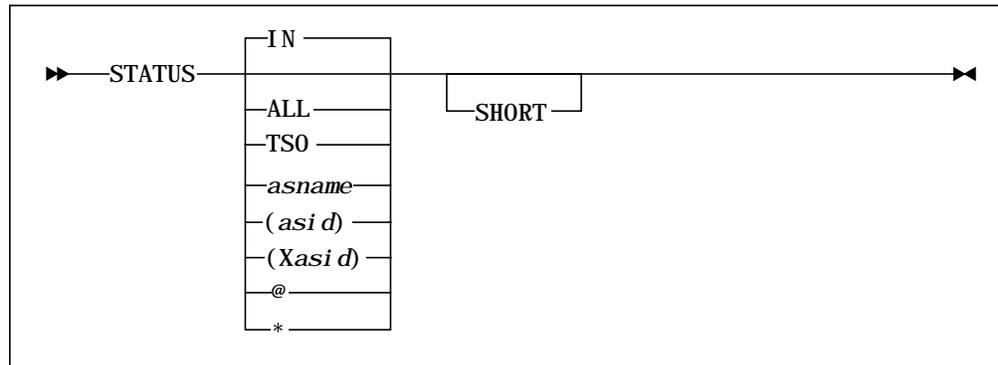
Note: The appearance of *** means that these values are temporarily invalid because they are being updated by SRM. Invoke the SRM service again to display updated values.

5. Contents of the SSCVTSUSE field for this subsystem.

STATUS

The STATUS (STA) service monitors and displays the current status of active address spaces in the system.

Syntax



where

IN	Displays the status of all address spaces swapped in memory; the default.
ALL	Displays the status of all address spaces.
TSO	Displays the status of all TSO users.
<i>asname</i>	Displays the status of a specific address space name.
(<i>asi d</i>)	Is the address space identifier in decimal format.
(<i>Xasi d</i>)	Is the address space identifier in hexadecimal format.
@	Displays the status of your own address space.
*	Displays the status of the last address space entered.
SHORT	Shortens the display to accommodate console screen width.

Examples

To display the status of all active address spaces swapped in memory, type:

```

status
      ①      ②      ③ ④ ⑤
AMTS21I INVENTORY ( 6) IN CP FM( 35) WS( 16) CPU 1234.56 Q(09) PGP 10/1
      ⑥      ⑦      ⑧      ⑨
AMTS21I SMIGO ( 47) NSW LW FM(215) WS(***) CPU 168.12 Q(01) PGP 9/3
AMTS21I *MASTER* ( 1) NSW FM( 97) WS(***) CPU 21.59 TRQ PGP 0/XX
AMTS21I PCAUTH ( 2) NSW FM( 35) WS(***) CPU 0.03 TRQ PGP 1/ 1
AMTS21I RASP ( 3) NSW FM( 76) WS(***) CPU 0.09 TRQ PGP 1/ 1
AMTS21I TRACE ( 4) NSW FM( 98) WS(***) CPU 0.01 TRQ PGP 1/ 1
AMTS21I XCFAS ( 5) NSW FM(174) WS(***) CPU 0.34 TRQ PGP 1/ 1
AMTS21I GRS ( 6) NSW FM(151) WS(***) CPU 0.06 TRQ PGP 1/ 1
AMTS21I SMXC ( 7) NSW FM( 16) WS(***) CPU 0.00 TRQ PGP 10/ 1
AMTS21I SYSBMAS ( 8) NSW FM( 22) WS(***) CPU 0.00 TRQ PGP 1/ 1
AMTS21I DUMPSRV ( 9) NSW FM( 29) WS(***) CPU 1.05 TRQ PGP 5/ 1
AMTS21I CONSOLE ( 10) NSW FM(155) WS(***) CPU 22.46 TRQ PGP 1/ 1
    
```

Legend:

1. Address space name.
2. Address space ID (ASID).
3. Status indicator #1. Possible values are as follows:

Value	Description
NSW	Nonswappable
LS	Logically swapped
PVL	Privileged
OUT	Swapped out of memory
GOI	Going in
GOB	Going between states
ENQ	Enqueue privileged
IN	In memory
G00	Going out of memory

4. Status indicator #2. Possible values are as follows:

Value	Description
LS	Address space is logically swapped-out
MS	Main storage optimized detected long wait
LW	Long wait (explicit)
CP	Significant CPU user

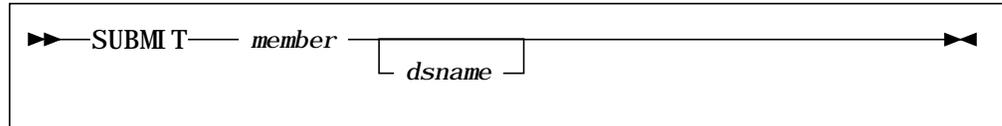
5. Real frames currently in use by this address space.
6. Current working set in pages. If a job is nonswappable, the working set is displayed as ***. The current working set is the number of page frames swapped at the last swap action. This is also the number of page frames necessary to swap in an address space.

7. CPU time used by the current step in this address space, in seconds and hundredths.
8. TRQ indicates that the address space is on the truly ready queue; NTRQ indicates that the address space is not on the truly ready queue.
9. Performance group and period for this address space. An asterisk (*) indicates the value is temporarily invalid.

SUBMIT

The SUBMIT (SUB) service submits a job from the specified data set to the job entry subsystem. This service enables you to submit JCL for routine operations jobs. Because the JCL contains a JOB statement, a valid user account code may be used for accounting purposes. The job runs as a problem program; consequently, SMF data can be obtained.

Syntax



where

member Is a member in the BBPARM library data set.

dsname Is the name of the data set containing the member to be submitted. If a data set name is not specified, the data set last specified on a SUBMIT command is used. Therefore, if you are submitting multiple members from the same data set, you only need to specify the data set name on the first SUBMIT command. If a data set name has not been specified and DDNAME BBPARM was preallocated, that data set will be used.

Example

To submit BBPARM library member *backup* to the job entry subsystem, type:

```

submit backup
AMTV11I MEMBER BACKUP SUBMITTED TO JOB ENTRY SYSTEM
  
```

To submit the job(s) in member *backup* in data set MY.PARMLIB, type:

```

submit backup, my. parmlib
  
```

If the above data set name was previously entered, that previous data set MY.PARMLIB will be used. If a data set was not previously specified and DDNAME BBPARM was preallocated, that data set will be used. To submit the job(s) in member *backup2*, type:

```

submit backup2
  
```

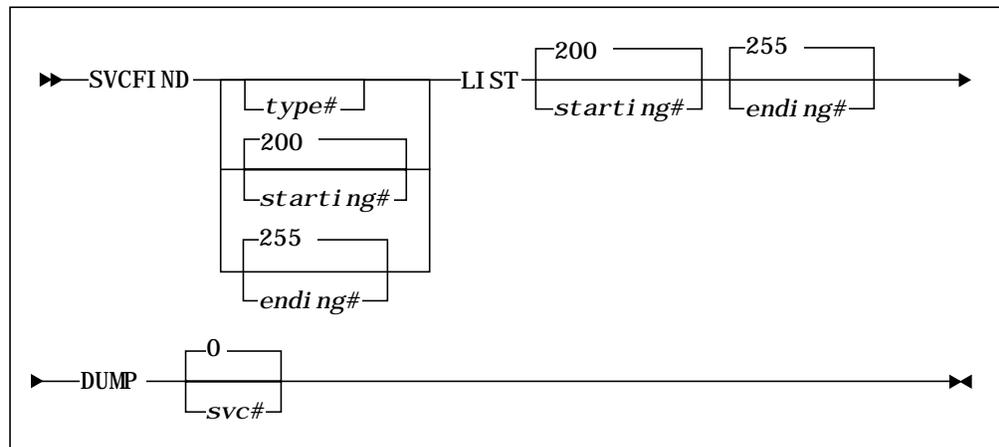
SVCFIND

The SVCFIND (SVC) service

- Scans the SVC table, locating and displaying information about SVCs.
- Lists unused SVCs.
- Dumps the first 64 bytes of the SVC module from main storage.
- Helps you plan the installation of new SVCs.

For information on loading SVCs into main storage, see [“LOADLPA” on page 124](#).

Syntax



where

<i>type#</i>	Is a 1-digit SVC type number (1, 2, 3, 4, or 6). SVCFIND displays unused SVC table entries that equal the type number.
<i>starting#</i>	Is a 1- to 3-digit SVC number. The display begins with this number. The default is 200.
<i>ending#</i>	Is a 1- to 3-digit SVC number. The display ends with this number. The default is 255.
LIST	Displays all SVC numbers in the specified starting number and ending number range.
DUMP	Displays the first 64 bytes of the specified <i>svc#</i> from main storage.
<i>svc#</i>	Is a 1- to 3-digit SVC number; this number cannot exceed 255. The default is 0.

Note: If no operands are specified, SVCFIND displays all unused SVC table entries, regardless of type number. This is the default.

Examples

To display SVC table entries for unused Type 2 SVC numbers between 240 and 245, type:

```

svcfind 2 240 245

AMTY11I UNUSED SVCTABLE ENTRIES 240-245 TYPE=2
          ① ① ① ① ② ③ ④
AMTY13I SVC# TYPE EPA      APF NP  ESR ASF LOCKS TABLE-ADR FLAGS
AMTY14I 240 2   -UNUSED-  N  N   N  Y  NONE 00FDB4A0 81000000
AMTY14I 243 2   -UNUSED-  N  N   N  Y  NONE 00FDB4B0 81000000
    
```

To display all SVC types, used and unused, between 199 and 201, type:

```

svcfind list 199 201

AMTY11I SVCTABLE ENTRIES 199-201
          ① ① ① ① ② ③ ④
AMTY13I SVC# TYPE EPA      APF NP  ESR ASF LOCKS TABLE-ADR FLAGS
AMTY14I 199 2   -UNUSED-  N  N   N  Y  NONE 00FDB358 81000000
AMTY14I 200 2   -UNUSED-  N  N   N  Y  NONE 00FDB360 81000000
AMTY14I 201 2   00E1AC10  Y  N   N  N  NONE 00FDB368 C8000000
    
```

To dump the first 64 bytes of storage for SVC 235, type:

```

svcfind dump 235

AMTY15I SVC 235 DISPLAY
          ⑤ ⑥
AMTY16I 81AF6000 47F06046 41D9C5E2 D6D3E5C5 61E7C160 *. 0- . . RESOLVE/XA- *
AMTY16I 01AF6010 E2E3D660 E2E5C360 F0F861F1 F261F8F6 *TSO- SVC- 08/15/92*
AMTY16I 01AF6020 60F1F84B F2F44DC3 5D40F1F9 F8F640C2 *- 11. 24(C). 1992. B*
AMTY16I 01AF6030 D6D6D3C5 40C1D5C4 40C2C1C2 C2C1C7C5 *00LE. AND. BABBAGE*
    
```

Legend:

1. Bit settings in the SVCTABLE entry. Refer to the IBM Publication, *MVS Diagnosis: Data Areas*, for a description of the SVCTABLE data area.
2. Highest lock required by the SVC first-level interrupt handler; the lock is obtained from the SVCLOCKS field of the SVCTABLE.
3. Address of the SVCTABLE entry for the appropriate SVC.
4. A hexadecimal display of the SVCATTR and SVCLOCKS fields from the SVCTABLE entry.
5. The starting address of the storage being dumped and the hexadecimal and EBCDIC representation of that data.
6. Text representation of the dump.

SYSDUMP

The SYSDUMP (SYS) service displays the title of the dump, as well as the date and time the dump was taken for each active dump data set. This service also determines if SVC dumps should be retained or deleted, which helps prevent the loss of important dumps due to all dump data sets being full. (See “CLEAR” on page 61 for information on clearing dump data sets.)

Syntax

```
▶▶—SYSDUMP —————▶▶
```

Example

To display information about each active dump data set, type:

```
sysdump
AMTDU0I  SYS1.1DUMP01  WAS FILLED AT 10:21 ON MAR 14,2 1996
AMTDU1I  SOURCE=3SVCDUMP
AMTDU2I  TITLE=4ERROR IN TSO
```

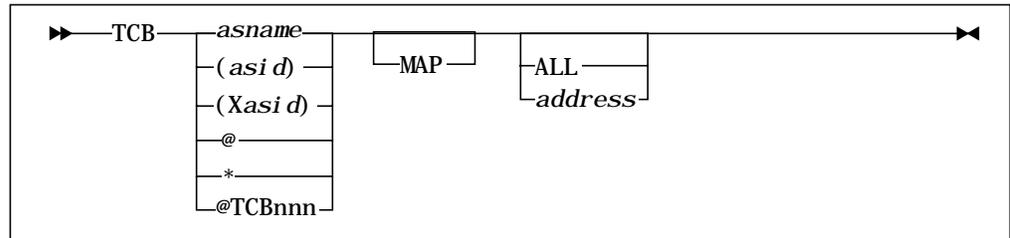
Legend:

1. A full-dump data set.
2. Time and date of dump.
3. System component that took the dump.
4. Message text describing the dump.

TCB

The TCB service displays internal values of task control blocks (TCBs) and request blocks (RBs) belonging to an address space.

Syntax



where

- asname* Is the address space name.
- (asi d)* Is the address space identifier in decimal format.
- (Xasi d)* Is the address space identifier in hexadecimal format.
- @ Specifies your own address space.
- * Specifies the last address space entered.
- @TCBnnn Is the symbol generated by the TCB service when one of the other parameters is used by a prior invocation. One symbol is generated for each TCB within the target address space, and the symbols are numbered consecutively beginning with 001.

The symbols are deleted and reassigned each time the TCB service is invoked specifying any value other than @TCBnnn as the first parameter.
- MAP Causes the general registers from the request block to be displayed.
- ALL Causes all TCBs associated with the requested address space to be displayed.
- address* Is the hexadecimal address of the current step or subtask TCB.

Example

To display the internal TCB and RB values for address space INVENTORY, type:

```

tcb inventory map
AMIT71I ADDRESS SPACE: INVENTORY STEP: APPC PRTY: 175 TCBS: 4
AMIT72I # 1 TCB 7FE158 PRTY 255 NDSP 0400 PK 00 JOBSTEP
AMIT7LI -----
AMIT7YI OTC 000000 NTC 000000 LTC 7FF7C0
AMIT73I PRB 007FDC58 EP 833B1538 NM IEAVAR00
AMIT7XI PSW 070C1000 810F86B0 WT 1 CDE 00F4D8C8
AMIT74I 0- 3 00000001 FF801B54 007FE410 00FD1FF0
AMIT74I 4- 7 007FE158 007FDC58 833B155E 00F60400
AMIT74I 8- 11 00000000 833B15E8 010F8678 007FE4AC
AMIT74I 12- 15 833B2090 007FE410 833B20E6 807FDC58

AMIT72I . # 2 TCB 7FF7C0 PRTY 255 NDSP 0400 PK 80
AMIT7LI . -----
AMIT7YI . OTC 7FE158 NTC 7FFAC8 LTC 7F3D20
AMIT73I . PRB 007FF430 EP 00E53000 NM IEESB605
AMIT7XI . PSW 070C1000 80E53706 WT 0 CDE 00C0DFA8
AMIT74I . 0- 3 00000028 007FC1BC 007FD148 807FD0E8
AMIT74I . 4- 7 00000004 807FD11C 056B7980 0000000C
AMIT74I . 8- 11 007FD0E8 007FD008 007FC0D0 00E5302C
AMIT74I . 12- 15 00E53CB4 007FC1C0 00000028 007FC1A8
AMIT73I . PRB 007FF330 EP 80E3F5A8 NM IEFSD060
AMIT7XI . PSW 070C1000 80E47CE2 WT 1 CDE 00C07DD8
AMIT74I . 0- 3 00000001 FF800DB0 856B79BC 7FFDFE78
AMIT74I . 4- 7 007F2858 007F2A58 007F3DDC 007ED268
AMIT74I . 8- 11 007FF238 807F37A8 00F60400 007F3D20
AMIT74I . 12- 15 00E473C2 007F28C0 80E47C24 807FF330

AMIT72I . # 3 TCB 7F3D20 PRTY 255 NDSP 0400 PK 80
AMIT7LI . -----
AMIT7YI . OTC 7FF7C0 NTC 000000 LTC 7F9320
AMIT73I . PRB 007F3C20 EP 827C9000 NM IKJEFT01
AMIT7XI . PSW 071C1000 827CDE1C WT 1 CDE 00C0C2C0
AMIT74I . 0- 3 00000001 FF81A5EC 827CD3D8 027CE3D7
AMIT74I . 4- 7 807E5F40 00000000 007E5FE8 007FDED8
AMIT74I . 8- 11 007CDE38 007FF510 007E5F18 027CF3D6
AMIT74I . 12- 15 007E55D8 007E55D8 827CDE20 807F3C20

AMIT72I . # 4 TCB 7F9320 PRTY 255 NDSP 0400 PK 80
AMIT7LI . -----
AMIT7YI . OTC 7F3D20 NTC 7F9E88 LTC 7B44C0
AMIT73I . PRB 007F9220 EP 827D5000 NM IKJEFT02
AMIT7XI . PSW 078C1000 827D59A6 WT 1 CDE 00C0C338
AMIT74I . 0- 3 80000001 FFFB47AC 00F60480 00000038
AMIT74I . 4- 7 827D5142 0000003C 827D5046 027D6045
AMIT74I . 8- 11 007E5130 0004B828 027D7044 027D8043
AMIT74I . 12- 15 007B7E28 007B7E28 827D5142 807F9220
    
```

Legend:

1. Address space name (source: TIOCJOB).
2. Current stepname of this address space (source: TIOCSTEP).

3. Priority of the address space.
4. Total number of TCBs for the address space.
5. Number of the TCB being displayed.
6. Address and dispatching priority of the TCB.
7. Nondispatchability flags (TCBFLGS4 and TCBFLGS5).
8. Protect key.
9. TCB chaining pointers.
10. Type and address of each RB associated with this TCB and the entry-point address and name of the module. SVC numbers are given in decimal format; for example, SVC-18 corresponds to BLDL.
11. Current PSW in this RB.
12. Wait count of the RB.
13. CDE address.
14. Registers for the RB, which are displayed only when the MAP option is specified.

To display detailed information for the specific TCB referenced by the @TCBnnn symbol, type:

```

tcb, @tcb004

AMT001A RESOLVE PLUS

AMT71I ADDRESS SPACE: JES2      STEP: JES2      PRTY: 249 TCBS: 18

AMT72I # 4 TCB 7E09C0  PRTY 254  NDSP 0400  PK 10
AMT7LI -----
AMT7YI 0TC 7FD810 NTC 7E11D8 LTC 000000
AMT73I PRB 007E0938 EP 800D16F0                      NM HOSPOOL
AMT7XI PSW 071C0000 800D179C WT 1 CDE 007FDC90
AMT74I 0- 3 00000001 0010EA58 00045A38 0010E9D0
AMT74I 4- 7 00000000 000001D8 00006000 0010E9D0
AMT74I 8- 11 000E5718 00000000 800CEAF6 00006000
AMT74I 12- 15 800D16F0 0010E9D0 80FD84A8 807E0938
    
```

TERM

The TERM (TER) service displays information about partially terminated address spaces.

Syntax

```
▶▶—TERM—————▶▶
```

Example

To display address spaces in abnormal termination, type:

```
term
      ①          ②
AMTT11I  JOB INVENTORY FAILED CODE SOC7
AMTT11I  JOB UPDATE1  FAILED CODE U0001
```

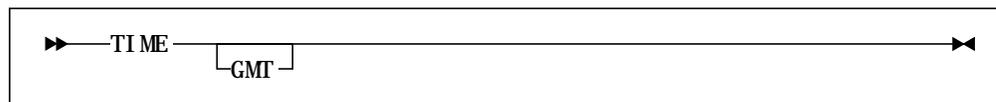
Legend:

1. Job that is in abnormal termination.
2. Abend code for failed job.

TIME

The TIME (TIM) service displays date and time. The date is displayed in both Gregorian and Julian formats. The Greenwich Mean Time (GMT) time display includes the offset to the local time.

Syntax



where

GMT Displays the Greenwich Mean Time specified by the operator at the time of IPL.

Examples

To display the current date and time, type:

```
time
AMIT32I  DATE 3/21/1996 (1996.081), local time 11:39:39.82
```

Legend:

1. Gregorian date.
2. Julian date.
3. Local time in hours, minutes, seconds, and hundredths of seconds.

To display the current date and GMT time, type:

```
time, gmt
AMIT32I  DATE 3/21/1996 (1996.081), GMT time 19:45:40.56 (Local -8:00:00)
```

Note: You may type the `time gmt` command with or without a comma.

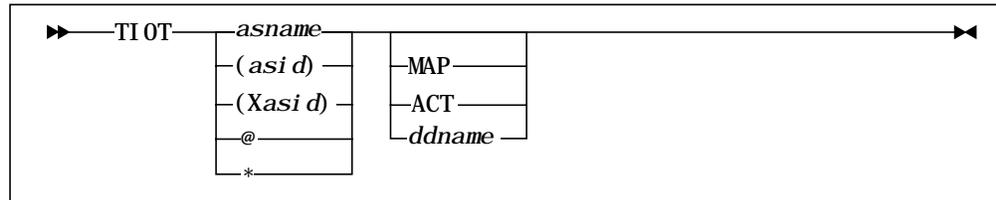
Legend:

1. Gregorian date.
2. Julian date.
3. Greenwich Mean Time.
4. Local time offset from Greenwich Mean Time.

TIOT

The TIOT (TIO) service displays all allocated devices for a specified address space and monitors the usage of data sets by address space.

Syntax



where

<i>asname</i>	Is an address space name.
(<i>asi d</i>)	Is an address space identifier in decimal format.
(<i>Xasi d</i>)	Is an address space identifier in hexadecimal format.
@	Specifies your own address space.
*	Specifies the last address space entered.
MAP	Displays the data set name(s) associated with each TIOT entry.
ACT	Displays the active TIOT entries that contain data sets with nonzero EXCP counts. The data set names are listed, as is the case with MAP.
<i>ddname</i>	Is the name of a DD statement. Specifying this parameter displays the data set names associated with the specified DD name.

Note: If, within a concatenation, more than one of the data sets are allocated to a common volume, TIOT attributes all the EXCPs for the data sets sharing that volume to the first data set within the concatenation that is on the current volume.

For data sets sharing a common volume within a concatenation, IBM updates the TCT I/O table only for the first data set in the concatenation that resides on that volume.

Examples

To display all allocated devices for address space INVENTORY and display the data set name associated with each TIOT entry, type:

```

tiot inventory map
AMTE11I JOB 332 INVENTORY UPDATE1 PRTY 279 PAGE I/O'S 1044
AMTE12I DD STEPLIB UNIT 151 VOLUME SYS001 EXCP 604
AMTE13I DSN SYS3.RESOLVE.LOAD
AMTE12I UNIT 250 VOLUME PERF01
AMTE13I DSN BB.CMC.CMC310.LOAD
AMTE12I UNIT 250 VOLUME PERF21 EXCP 129
AMTE13I DSN BB.CMC.CMC310.LINK
AMTE12I DD SYS00002 UNIT 254 VOLUME PUB001
AMTE13I DSN SYSCTLG.PUB001
AMTE12I DD SYS00003 UNIT 254 VOLUME PUB002 EXCP 15
AMTE13I DSN SYSCTLG.PUB002
    
```

Legend:

1. JES job ID.
2. Name of the specified address space.
3. Current stepname for the specified address space.
4. Internal dispatching priority for the address space.
5. Total page faults requiring a page I/O operation for this address space.
6. DD name.
7. Device number.
8. Serial number of volume mounted on device.
9. EXCP count for each data set within each DD name.
10. Data set name associated with TIOT entry; issued only if you specify MAP, ACT, or *ddname*.

To limit output to TIOT entries that have data sets with nonzero EXCP counts, type:

```

tiot inventory act
AMTE11I JOB 332 INVENTORY UPDATE1 PRTY 279 PAGE I/O'S 1044
AMTE12I DD STEPLIB UNIT 151 VOLUME SYS001 EXCP 604
AMTE13I DSN SYS3.RESOLVE.LOAD
AMTE12I UNIT 250 VOLUME PERF21 EXCP 129
AMTE13I DSN BB.CMC.CMC310.LINK
AMTE12I DD SYS00003 UNIT 254 VOLUME PUB002 EXCP 15
AMTE13I DSN SYSCTLG.PUB002
    
```

To limit output to the data set names associated with the DD name STEPLIB, type:

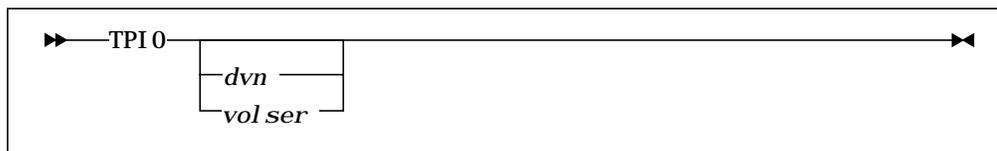
```
tiot inventory, steplib
```

```
AMTE11I JOB 332 INVENTORY UPDATE1 PRTY 279 PAGE I/O'S 1044  
AMTE12I DD STEPLIB UNIT 151 VOLUME SYS001 EXCP 604  
AMTE13I DSN SYS3.RESOLVE.LOAD  
AMTE12I UNIT 250 VOLUME PERF01  
AMTE13I DSN BB.CMC.CMC310.LOAD  
AMTE12I UNIT 250 VOLUME PERF21 EXCP 129  
AMTE13I DSN BB.CMC.CMC310.LINK
```

TPIO

The TPIO (TPI) service displays all outstanding I/O, including TP devices. This service also determines whether TP lines are enabled (have outstanding I/O).

Syntax



where

dvn Is the device number for which outstanding I/O is to be displayed.

vol ser Is the volume serial number for which outstanding I/O is to be displayed.

Examples

To display all outstanding I/O, type:

```

tpio
  ①  ②      ③      ④      ⑤
AMTI 11I  TSO      UNIT  021      IOQ  00F832A0  DRIVER- VTAM
AMTI 11I  INVENTORY UNIT  238 WORK01 IOQ  00F91300  DRIVER- EXCP
    
```

Legend:

1. Address space name.
2. Device number.
3. Volume serial number of the device, if DASD or a tape device.
4. IOQ address.
5. Name of IOS driver. This can be as follows:

Code	Driver
MISC	Miscellaneous driver
EXCP	EXCP driver
VSAM	VSAM driver
VTAM	VTAM driver
TCAM	TCAM driver
OLTEP	OLTEP driver

Code	Driver
PCIFETCH	Program FETCH driver
JES3	JES3 subsystem
MSS/MSC	Mass storage subsystem
IOSPURGE	Internal IOS PURGE IOQ routine
VPSS	Vector processing subsystem (3838 array processor)
CRYPTO	Cryptographic subsystem
ASM	Auxiliary storage manager (paging supervisor)
DYNPATH	Path reconfiguration
SVC-33	I/O HALT I/O SVC routine
R-CLEAR	Clear device recovery
R-SUBCHN	Subchannel recovery
SVCPURGE	I/O PURGE I/O SVC routine
ALTPATH	Alternate path recovery
MIH	Missing interrupt handler
*UNKNOWN	Unknown or unassigned driver code in use

To display all outstanding I/O for device 283, type:

```

tpio 283
      ①      ②      ③      ④
AMTI 31I  STC    205 G0JOB  STEP1  PRTY 7(90) PGP 1/2
AMTI 32I  UNIT   283 PACK08 IOQ 003F88 IOSB 003F00 DVR EXCP ⑤
AMTI 33I  QDEPTH 5 RSV  CNT 0 USE  CNT 3 DCB  CNT 2
      ⑥      ⑦      ⑧      ⑨

```

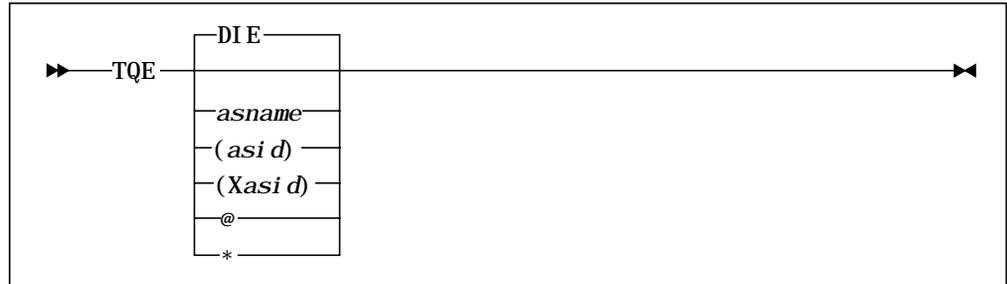
Legend:

1. JES job ID of the address space using the device.
2. Name and current step for the address space using the device.
3. Dispatching priority for the address space.
4. Performance group and period.
5. Unit, volume serial number, IOQ and IOSB addresses of the I/O, and the I/O driver for this operation.
6. Number of I/Os waiting for access to this device.
7. Number of address spaces that have requested a reserve on this device.
8. Number of address spaces that have allocated this device.
9. Number of open DCBs against this device.

TQE

The TQE service displays information about Disabled Interrupt Exit (DIE) routines scheduled for execution and pending timer interrupts for an address space.

Syntax



where

- | | |
|-----------------|--|
| DIE | Indicates Disabled Interrupt Exit. This operand displays the TQE chain for all DIEs; the default. |
| | Note: A DIE executes as an extension to the first-level interrupt handler. Therefore, a DIE routine interrupts and executes under any dispatchable unit of work (TCB or SRB) for any address space. |
| <i>asname</i> | Is an address space name. |
| <i>(asi d)</i> | Is an address space identifier in decimal format. |
| <i>(Xasi d)</i> | Is an address space identifier in hexadecimal format. |
| @ | Specifies your own address space. |
| * | Specifies the last address space entered. |

Examples

To display the TQE chain for all DIEs, type:

```

tqe
AMTQE1I TIMER QUEUE ELEMENTS, ACTIVE AT 11:16:45.59
AMTQE2I EXPIRE TCB
AMTQE3I TIME JOBNAME ADDRESS TYPE FLAGS CREATOR
      ①      ②      ③      ④      ⑤      ⑥
AMTQE4I 11:16:45.60 ----- REAL SRM
AMTQE4I 11:16:45.76 DCSPAS ----- REAL USER SETDI E
AMTQE4I 11:16:45.77 DCSRES ----- REAL USER SETDI E
AMTQE4I 11:16:45.82 *MASTER* ----- REAL USER SETDI E
AMTQE4I 11:16:45.91 DCSPAS ----- REAL USER SETDI E
AMTQE4I 11:16:45.94 XCFAS ----- REAL USER SETDI E
AMTQE4I 11:16:48.46 LRLMPROC ----- REAL USER SETDI E
AMTQE4I 11:17:29.33 JES2 ----- REAL USER SETDI E
AMTQE4I 11:17:44.09 ----- REAL TLI M
AMTQE4I 11:17:47.36 ----- REAL USER SETDI E
AMTQE4I 00:00:00.00 ----- REAL MI DN

```

Legend:

1. Expiration time for the TQE.
2. Jobname for the TQE.
3. TCB address for the TQE; a value is displayed only if the TYPE field contains TASK or WAIT.
4. Type of TQE; can be REAL, TASK, WAIT, or UNKN.
5. Flags for the TQE; flags can be one of the following:

REAL	Real TQE being timed
USER	Non-system TQE
DUMY	Dummy system TQE
TLI M	Time limit checking system TQE
SRM	System Resources Manager TQE
RMF	RMF system TQE
MI DN	Midnight system TQE
UNKN	Unknown TQE
6. Macro that created the TQE; it can be SETDI E, STI MER, STI MERM, or blank.

To display TQEs for address space JES2, type:

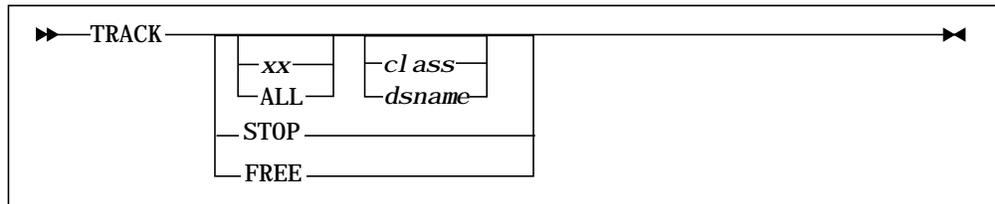
```
tqe jes2
```

```
AMTQE1I  TIMER QUEUE ELEMENTS, ACTIVE AT 11: 19: 18. 10
AMTQE2I  EXPIRE                                TCB
AMTQE3I  TIME          JOBNAME      ADDRESS      TYPE      FLAGS  CREATOR
AMTQE4I  11: 19: 18. 13 JES2        007F0648    WAIT     USER   STIMER
AMTQE4I  11: 19: 18. 13 JES2        -----    REAL     USER   SETDI E
AMTQE4I  11: 19: 22. 45 JES2        007F03B8    WAIT     USER   STIMER
AMTQE4I  11: 19: 29. 33 JES2        -----    REAL     USER   SETDI E
```

TRACK

The TRACK (TRA) service provides a record of services invoked during the session and writes service output to a data set or SYSOUT.

Syntax



where

xx	Is a user-specified suffix. You can track up to 32 services by placing the service names in a member in the BBPARM library data set. The member is named \$\$INTKxx, where xx is the user-specified suffix.
class	Specifies the SYSOUT class to which tracking output is routed.
dsname	Allows specification of a data set name to which output is routed.
ALL	Initiates tracking for every user-invoked service.
STOP	Terminates tracking.
FREE	Deallocates tracking output. If routed to SYSOUT, tracking output is passed to JES for immediate printing.

Examples

To initiate tracking of services specified in member \$\$INTK24 in the data set myi d.output, type:

```
track 24, myi d. output
```

The output produced by the tracked services will be recorded in data set myi d. output.

Note: A TRACK command may be included in the \$\$INTKxx member to specify the name of the output data set. If \$\$INTKxx contains a TRACK command, the first parameter is ignored if it is anything other than ALL. (Nesting is not permitted.)

To initiate tracking of services specified in member \$\$INTK01, type:

```
track, 01, A
```

The output will be written to SYSOUT class A.

To initiate tracking of all services and record the output in data set myi d.output, type:

```
track, ALL, myi d. output
```

To stop tracking and free an output data set, type:

```
track, FREE
```

To stop tracking (leaving an output data set allocated), type:

```
track, STOP
```

\$\$INTKxx Control Statements

A \$\$INTKxx member is read from the BBPARM data set under two conditions:

1. During initialization, when \$\$INTK00 is read and processed if present.
2. Whenever a two-character suffix (xx) is specified as the first parameter on a TRACK command.

A \$\$INTKxx member may contain a TRACK command and/or a list of up to 32 services that are to be tracked. Each service must be listed on a separate line beginning in column 1.

When the \$\$INTKxx member contains a TRACK command, the second parameter specifies the SYSOUT class or data set name for the tracking output. If the first parameter is a suffix, it is ignored. If the first parameter is ALL, all services are tracked.

Syntax Rules

1. List each service on a separate statement.
2. Begin each entry in column 1.
3. Enter up to 32 services to be tracked.

Examples

The following statement in \$\$INTK00 will cause all services to be tracked and the tracking output to be written to a SYSOUT class T data set.

```
TRACK, ALL, T
```

The following statements will cause only the CSA and RSM services to be tracked. The tracking output will be written to data set MY.TRACK.DS. The TRACK command may be on any line. If multiple TRACK commands are present, only the last one will be used.

```
CSA  
RSM  
TRA, xx, MY. TRACK. DS
```

Usage Notes

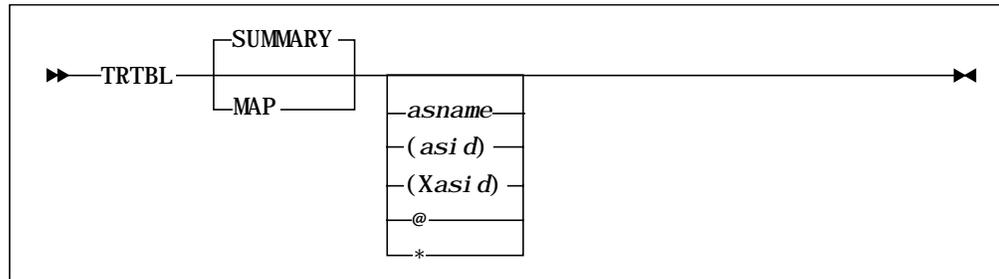
- Tracking provides a record of command input and output. You may request tracking for all services or up to 32 specific services. You may direct the tracking output to a SYSOUT data set or an existing cataloged data set.
- The tracking facility is controlled by the TRACK service. See [“TRACK” on page 234](#) for a description of this service.
- Tracking output is formatted with headings and timestamps. The output format is VBA and the logical record length is 137 bytes.

- Tracking output is controlled by the second parameter on the TRACK command or by the TRACK DD statement if present. By default, output is written to SYSOUT class A.

TRTBL

The TRTBL (TRT) service displays and summarizes data contained in the system trace buffers. This service provides detailed trace data from either a specific address space or the entire system.

Syntax



where

- SUMMARY** Displays summary counts of trace table entries; the default.
- MAP** Formats each trace table entry.
- asname* Is the address space name.
- (asi d)* Is the address space identifier in decimal format.
- (Xasi d)* Is the address space identifier in hexadecimal format.
- @ Specifies your own address space.
- * Specifies the last address space entered.

Examples

To summarize all system activity recorded in the trace buffers, type:

trtbl										
AMIT29I	¹ TYPE	² COUNT	TYPE	COUNT	TYPE	COUNT	TYPE	COUNT	TYPE	COUNT
	SRB	29	DSP	23	SVCR	90	SVC	88	PC	8
	PT	6	SSCH	20	WAIT	23	I/O	22	PGM	2

Legend:

1. Trace table entry (TTE) type. Possible trace entry types are as follows:

- CLKC Clock comparator interrupt occurred
- DSP Task dispatch
- I/O I/O interrupt received from I/O subsystem
- PC Program-call instruction executed

Usage Note

The trace table usually contains thousands of entries. The TRTBL service can display many screens of output. To reduce the number of screens, use the address space operand.

5. Terminal name.

To display information about TSO user TSO1, type:

```
tsulist tso1  
AMTTS1I  USERID  ASID    LINE  SYS  NODE  
AMTTS2I  TS01      17     000  VTAM LSNAB05  
AMTTS3I  8 USERS ARE LOGGED ON
```

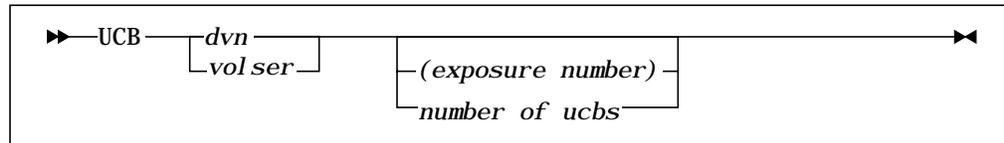
To display information about TSO users whose user IDs begin with the letters TSO, type:

```
tsulist tso*  
AMTTS1I  USERID  ASID    LINE  SYS  NODE  
AMTTS2I  TS01      17     000  VTAM LSNAB05  
AMTTS2I  TS02      49     000  VTAM LSNAC28  
AMTTS3I  8 USERS ARE LOGGED ON
```

UCB

The UCB service displays a unit control block (UCB) in hexadecimal and character formats. In addition, important information contained in the UCB, such as device type, volume serial status, and path definition, is interpreted and presented in messages AMTU2AI, AMTU2BI, and AMTU2CI. Either a device number or volume serial number may be specified.

Syntax



where

- | | |
|-----------------------|--|
| <i>dvn</i> | Is the device number whose UCB you want to display. |
| <i>vol ser</i> | Is the volume serial number you want to display. |
| <i>number of ucbs</i> | Specifies the number of UCBs displayed, beginning with the specified device. |

Example

To display the unit control block for volume serial number BAB320 type:

```

ucb bab320      ❶
AMTU2AI DVN=30B, Device class=DASD, Type=3390, Online, allocated ❷
AMTU2BI Volser=BAB320, Status=Private, Shared ❷
AMTU2CI Defined paths=2, Active paths=2, LCU No= 23 ❷

AMTU21I UCB 30B          RELOCATED PREFIX ❸
AMTU2DI                (First field is UCBRSTEM)
AMTU25I 01F10BA8 +0464 +X01D0 00040040 00000000 *.....* ❸
AMTU25I 01F10BB0 +0472 +X01D8 00000000 000106B0 *.....*
AMTU25I 01F10BB8 +0480 +X01E0 289C008C C00080C0 *.....*
AMTU25I 01F10BC0 +0488 +X01E8 2620FFFF FFFFFFFF *.....*
AMTU25I 01F10BC8 +0496 +X01F0 01080000 00000001 *.....*

AMTU21I UCB 30B          PREFIX ❹
AMTU25I 00F3F500 -0008 -X0008 00000000 00FBC600 *.....F.*

AMTU22I UCB 30B          COMMON SECTION ❺
AMTU25I 00F3F508 +0000 +X0000 008AFF8C 030B0000 *.....*
AMTU25I 00F3F510 +0008 +X0008 00F3F588 00F3FOC2 *.35h.30B*
AMTU25I 00F3F518 +0016 +X0010 3030200F 00F3F4E0 *.....34.*

AMTU23I UCB 30B          DEVICE DEPENDENT SECTION ❻
AMTU25I 00F3F520 +0024 +X0018 61D10100 C2C1C2F3 *.J..BAB3*
AMTU25I 00F3F528 +0032 +X0020 F2F01065 00000082 *20....b*
AMTU25I 00F3F530 +0040 +X0028 30688076 00BC1B20 *.....*

AMTU24I UCB 30B          COMMON EXTENSION ❼
AMTU25I 00F3F4E0 +0000 +X0000 00000940 20A20000 *.....s.*
AMTU25I 00F3F4E8 +0008 +X0008 01F10BA8 00000065 *.1.y....*
AMTU25I 00F3F4F0 +0016 +X0010 00000000 00FCCD3C *.....*
AMTU25I 00F3F4F8 +0024 +X0018 00F3F530 00000000 *.35....*
    
```

Legend:

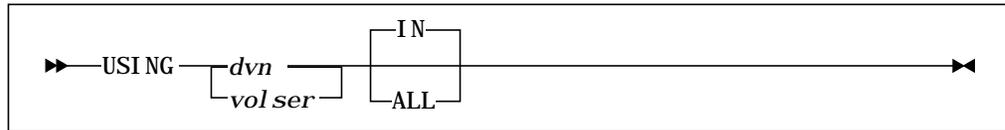
1. Specified device or volume serial number.
2. Interpreted information from UCB.
3. UCB relocated prefix section. The relocated prefix is a data area in the portion of the UCB prefix relocated above the 16MB line.
4. UCB prefix section for the device.
5. UCB common section for the device.
6. UCB device-dependent section for the device. The length of this section is dependent upon the type of device represented by the UCB.
7. UCB common extension for this device. The length of this section is dependent upon the type of device represented by the UCB.
8. Address in hexadecimal format.
9. Offset from the origin in decimal and hexadecimal format.

10. Hexadecimal representation of data (8 bytes per line).
11. Character representation of data (8 bytes per line).

USING

The USING (USI) service displays the active address spaces that are using a specified device. For example, if a tape drive appears inactive, you can determine which address space has allocated the device.

Syntax



where

- dvn* Is the hexadecimal device number.
- vol ser* Is a complete six-character volume serial number.
- IN Displays only the swapped-in address spaces using the device; the default.
- ALL Displays all the active address spaces using the device.

Example

To display the active address spaces using device BAB329, type:

```

using BAB329

13: 01: 47 CMD=USING BAB329

      ①      ②      ③
AMFD29I Unit Job Name DDNAME Data Set Name
AMFD27I D3F MQMPASRK STEPLIB MQS12. BASE. LINKLIB
AMFD27I D3F MQMPASRK BBILOAD MQS12. BASE. LINKLIB
AMFD27I D3F MQMPASRK ISPPROF BI TRKK. XDC2. ISPPROF
AMFD27I D3F DDH1X BBIPROF DDH1. BBPROF
AMFD27I D3F DDH1X IPCSDDIR IPCS. DDH1ESA. IPCSDDIR
AMFD27I D3F X18HGBG DFSOLP03 IMS. V5100. OLPO3
AMFD27I D3F X18HGBG MATRIBX IMS. V5100. MATRIBX
AMFD27I D3F MMQJB120 STEPLIB MQS12. BASE. LINKLIB
AMFD27I D3F MMQJB120 BBILOAD MQS12. BASE. LINKLIB
AMFD27I D3F MMQJB120 DBGLIB BBM33. ENG. DBGLIB
AMFD27I D3F BMVSLK2 BBCLIB BB. XTSTH. BBCLIB
AMFD27I D3F BMVSLK2 BBIPROF BMVSLK. BMVSLK2. BBPROF
AMFD27I D3F AAOSSDHC BBIPARM DDH1. DH31. TRAVEL
AMFD27I D3F BOLGBG3 STEPLIB IPX11. BOLGBG. LOAD
    
```

Legend:

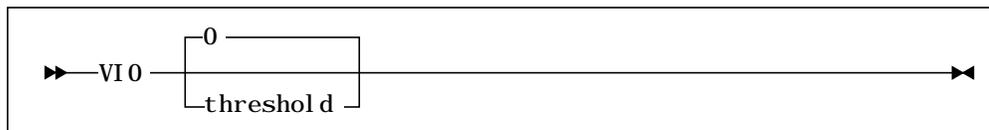
1. Job name.
2. DD name of the DD statement for the data set allocated to this device.

3. Data set name.

VIO

The VIO service displays usage of VIO slots by jobname or user ID.

Syntax



where

threshold Is a one- to four-digit integer; only address spaces that use a number of VIO slots greater than this value are listed; the default is 0.

Examples

To display the number of VIO slots used by each jobname or user ID, type:

```

vio
      ①
AMTVI 1I  VMON1   HAS      2 VIO SLOT(S)  IN USE
AMTVI 1I  SUP6    HAS      5 VIO SLOT(S)  IN USE
AMTVI 1I  TESTJB2 HAS     12 VIO SLOT(S)  IN USE
AMTVI 1I  VMON8   HAS      8 VIO SLOT(S)  IN USE
AMTVI 1I  VMON9   HAS      2 VIO SLOT(S)  IN USE
    
```

Legend:

1. Job name or user ID.
2. Number of VIO slots used.

To display user IDs or jobnames using 45 or more VIO slots, type:

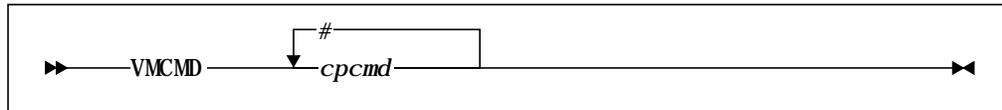
```

vio 45
AMTVI 3I  NO JOBS USING      45 OR MORE VIO SLOT(s)
    
```

VMCMD

The VMCMD (VM) service lets an OS/390 user running as a VM guest issue CP commands through the virtual console interface.

Syntax



where

cpcmd Is any valid CP command. A pound sign (#) delimits multiple CP commands.

Example

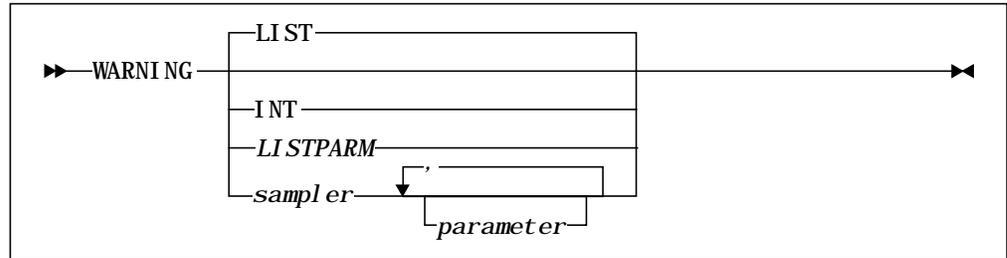
To enter the CP command Q NAMES, type:

```
vmcmd q names
```

WARNING

The WARNING (WARN) service controls the operation of the Exception Monitor (formerly Advanced Early Warning Services or AEWS) system. This service also monitors activity in the system and issues warning messages when user-specified thresholds are exceeded.

Syntax



where

- LIST** Lists the Exception Monitor samplers currently active; the default.
- INT** Modifies the report interval and sample interval for all active samplers based on what you specify in the INT field:
ss is a value between 05 and 60.
mmmm is a value between 100 and 1000.
- LISTPARM** Lists the currently active samplers and the number of samples taken.
- sampler** Lists the parameters currently in effect for that Exception Monitor sampler. A prompt is issued to allow you to modify these parameters, if desired.
- parameter** Is a parameter associated with the specified Exception Monitor sampler. Only the minimum number of letters necessary to identify the parameter need be specified.

Examples

To list the active Exception Monitor samplers, type:

```

aw list

AMTTH2I THE FOLLOWING PWS SAMPLERS ARE ACTIVE:
AMTTH3I GBL      REP
    
```

To change the interval in effect for the REP sampler (if it is active), type:

```

aew rep

AMITL1I REP  PARAMETER VALUE(S) :
AMITL2I     INTERVAL = 1
AMITL2I     TIME     = 1
AMITH9I ENTER NEW VALUES, NEXT, LIST, OR END
int=5
AMITH9I ENTER NEW VALUES, NEXT, LIST, OR END
end

```

To modify any or all parameters associated with an Exception Monitor sampler, type the values in the following format:

```

parameter=(value1, value2, . . .), parameter=(value1, value2, . . .), . . .

```

Note: When specifying a single value, parentheses are unnecessary.

To list a specific parameter for an Exception Monitor sampler, type the parameter name after the sampler name. Parameter names must be separated by commas. For example, to list only the INTERVAL and RPTNO parameter values for the BCPU sampler (if it is active), type:

```

aew bcpu, i, r

AMITL1I BCPU PARAMETER VALUE(S) :
AMITL2I     INTERVAL = 1
AMITL2I     RPTNO   = 5
AMITH9I ENTER NEW VALUES, NEXT, LIST, OR END

end

```

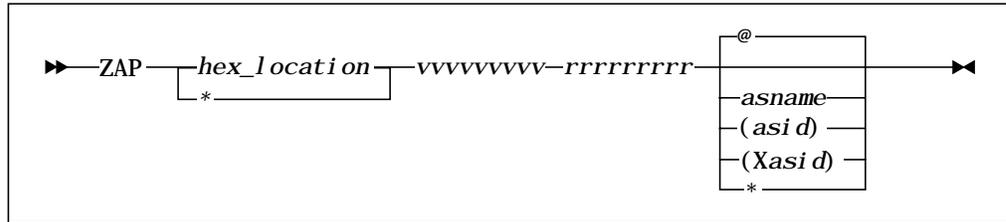
Usage Note

When a parameter or parameters have been successfully modified, the message ENTER NEW VALUES, LIST, OR END reappears. You can suppress this prompt before it appears by typing END after the last new parameter value.

ZAP

The ZAP service modifies main storage.

Syntax



where

- hex_location* Is the address field. This field must not exceed eight hexadecimal characters.
- ** Causes the ZAP service to modify the hex location used by the last invocation of the ZAP or DUMP service. The hex location the ZAP service modifies becomes the current hex location.

Optionally, the asterisk can be followed by an *offset*. When specified, an offset is a 1- to 4-digit hex value preceded by a plus or minus sign. This value indicates the relative distance from the beginning of the location defined by the asterisk to the area in storage to be displayed.
- vvvvvvvv* Is the Verify data field. You can change up to 6 bytes of data at a time. Specify the data in an even number of hex digits, from 2 to 12. You must verify all replaced data.
- rrrrrrrr* Is the Replacement data field. You can change up to 6 bytes of data at a time. Specify the data in an even number of hex digits, from 2 to 12. The Replacement field must be equal in length to the Verify field. You must verify all replaced data.
- asname* Is the address space name.
- (asi d)* Is the address space identifier in decimal format.
- (Xasi d)* Is the address space identifier in hexadecimal format.
- @* Specifies your own address space; the default.
- ** Specifies the last address space entered.

Example

To modify location 02F1C in the LLA address space from D2033008CA22 to 070007000700, type:

```
zap 02f1c d2033008ca22 070007000700 11a
```

```
AMTZ2CI REPLACE SUCCESSFUL
```

Usage Notes

- You can omit the *asname* parameter if the hex location is in common storage (CSA, LPA, SQA, or the nucleus).
- Changes made using the ZAP service are effective only for the duration of the IPL. The ZAP service page-fixes changes to hex locations in the following areas:

EPLPA	Extended Pageable Link Pack Area
EMLPA	Extended Modifiable Link Pack Area
MLPA	Modifiable Link Pack Area
PLPA	Pageable Link Pack Area

Changes to hex locations are page-fixed by ZAP to prevent the changes from being overwritten when page-ins occur.

- For processor complexes consisting of more than one CPU, you can specify a CPU number. Add /*c* to the end of the value specified for *hex location*, where *c* is the single hexadecimal number (0 through 9) of a specific CPU. Use this field when you ZAP the Prefixed Save Area (PSA).

Chapter 5. Exception Monitor Samplers

The Exception Monitor helps you detect potential problems by monitoring thresholds that you establish, and sending a warning message when a threshold is exceeded. Messages are sent both to the WARN view and to the OS/390 console.

To access the Exception Monitors:

1. Access SYSPROG services from the MainView for OS/390 ADMIN menu (or on the **COMMAND** line, type **SYSPROG**).
2. Select the WARNING service.

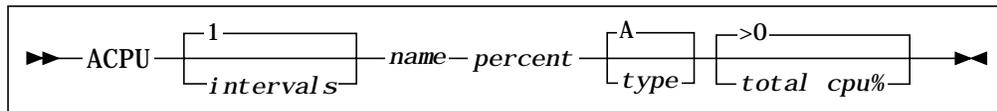
This chapter describes the samplers that MainView provides. Each sampler monitors a different aspect of system performance. You control the Exception Monitor by selecting samplers that are critical to your site, setting thresholds for them, and then building a control statement member with those samplers.

For information on how to create a control statement member, see the *MainView Customization Guide*.

ACPU

The Address Space CPU (ACPU) sampler sends a warning message to the operator when an address space uses more than a specified percentage of CPU time during the reporting interval. You can suppress the message if a given threshold for total system CPU utilization is not reached during a reporting interval.

Syntax



where

<i>interval s</i>	Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).
<i>name</i>	Is the name of the address space to be monitored. If the last character of the name is an asterisk (*), the name is to be treated as a generic address-space name prefix; that is, any address space whose name matches the character string up to but not including the asterisk is monitored.
<i>percent</i>	Is the CPU busy percentage for which an address space must be responsible before an exception message is issued for that address space.
<i>type</i>	Is the type of address space to be monitored, where <ul style="list-style-type: none"> B Indicates a batch job S Indicates a started task T Indicates a TSO user A Indicates any address space matching the specified name; the default
<i>total cpu %</i>	Is the percentage of system CPU busy that must be reached before exception messages are issued. The default is a system CPU busy percent greater than 0.

Example

Assuming a report interval of 30 seconds, this control statement:

```
acpu 4, *, 35, a, 50
```

displays warning messages at 2-minute intervals if the system-CPU busy percentage exceeds 50 percent over the preceding 2 minutes. Also, a warning is issued for any address space that has been active 35 percent of the total CPU-busy time.

Warning Message

The following warning message is issued by the ACPUsampler:

```
PWSACPU0 WARNING* CPU USAGE IS ppp% FOR ttt jjjjjjjj
```

Message PWSACPU0 indicates that the specified address space has exceeded the warning threshold for percentage of system CPU-busy time;

where

ppp Is the address space busy percentage.

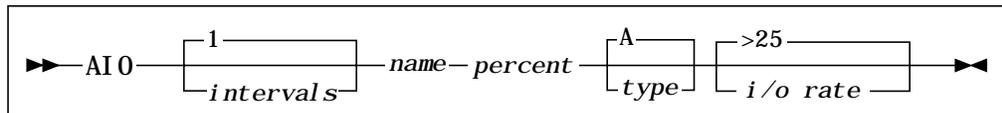
ttt Is STC, TSU, or JOB.

jjjjjjjj Is the address space name.

AIO

The Address Space I/O (AIO) sampler sends a warning message to the operator when an address space uses more than a specified percentage of the total system I/O activity during a reporting interval. You can suppress the message if a given total system I/O rate threshold is not reached during a reporting interval.

Syntax



where

<i>intervals</i>	Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).								
<i>name</i>	Is the name of the address space to be monitored. If the last character of the name is an asterisk (*), the name is to be treated as a generic address-space name prefix; that is, any address space whose name matches the character string up to but not including the asterisk is monitored.								
<i>percent</i>	Is the percentage of total system I/O for which an address space must be responsible before an exception message is issued for that address space.								
<i>type</i>	Is the type of address space to be monitored, where <table> <tr> <td>B</td> <td>Indicates a batch job</td> </tr> <tr> <td>S</td> <td>Indicates a started task</td> </tr> <tr> <td>T</td> <td>Indicates a TSO user</td> </tr> <tr> <td>A</td> <td>Indicates any address space matching the specified name; the default</td> </tr> </table>	B	Indicates a batch job	S	Indicates a started task	T	Indicates a TSO user	A	Indicates any address space matching the specified name; the default
B	Indicates a batch job								
S	Indicates a started task								
T	Indicates a TSO user								
A	Indicates any address space matching the specified name; the default								
<i>i/o rate</i>	Is the rate of EXCPs per second that must be reached before exception messages are issued. The default is a system I/O rate great than 25 EXCPs per second.								

Example

Assuming a report interval of 30 seconds, this control statement:

```
ai o 4, *, 35, a, 50
```

displays warning messages at 2-minute intervals if the system I/O rate exceeds 50 EXCPs-per-second over the preceding 2 minutes. Also, a warning is issued for any address space doing more than 35 percent of the I/O.

Warning Messages

The following warning messages are issued by the AIO sampler:

```
PWSAI000 WARNING* TOTAL SYSTEM I/O RATE IS ppppp EXCPS/SEC
```

Message PWSAIO00 indicates that the system-I/O rate has exceeded the warning threshold.

```
PWSAI001 WARNING* I/O RATE FOR ttt nnnnnnnn IS xxx% OF SYSTEM TOTAL
```

Message PWSAIO01 indicates that the specified address space has exceeded the warning threshold for percentage of system I/O;

where

ttt Is STC, TSU, or JOB.

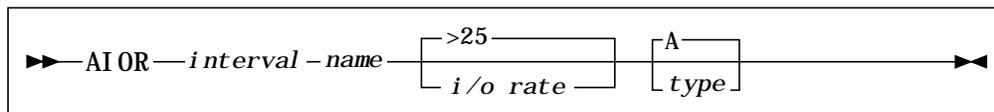
nnnnnnnn Is the address space name.

xxx Is the percent of the total system I/O rate.

AIOR

The Address Space I/O Rate (AIOR) sampler sends a warning message to the operator when an address space issues EXCPs at a rate that exceeds the defined threshold during a reporting interval.

Syntax



where

<i>interval</i>	Is the number of report intervals between warning messages.
<i>name</i>	Is the name of the address space to be monitored. If the last character of the name is an asterisk (*), the name is to be treated as a generic address-space name prefix; that is, any address space whose name matches the character string up to but not including the asterisk is monitored.
<i>i/o rate</i>	Is the rate of EXCPs-per-second that must be reached before exception messages are issued. The default is an I/O rate greater than 25 EXCPs per second.
<i>type</i>	Is the type of address space to be monitored, where
B	Indicates a batch job
S	Indicates a started task
T	Indicates a TSO user
A	Indicates any address space matching the specified name; the default

Example

Assuming a report interval of 30 seconds, this control statement:

```
ai or 4, *, 25, a
```

displays warning messages at 2-minute intervals for each address space whose I/O rate exceeds 25 EXCPs-per-second over the preceding 2 minutes.

Warning Message

The following warning message is issued by the AIOR sampler:

```
PWSAIR01 WARNING* I/O RATE FOR ttt nnnnnnnn IS ppp EXCPs/SEC
```

Message PWSAIR01 indicates that the specified address space has exceeded the warning threshold for I/O rate;

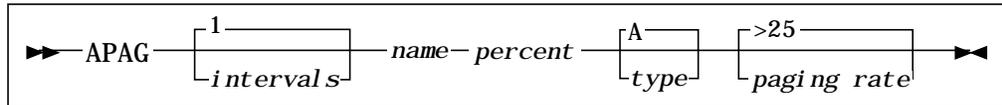
where

<i>ttt</i>	Is STC, TSU, or JOB.
<i>nnnnnnnn</i>	Is the address space name.
<i>ppp</i>	Is the number of EXCPs per second.

APAG

The Address Space Paging (APAG) sampler sends a warning message to the operator when an address space uses more than a specified percentage of the total system paging activity during a reporting interval. You can suppress the message if a given total system paging-rate threshold is not reached during the reporting interval.

Syntax



where

<i>interval s</i>	Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).								
<i>name</i>	Is the name of the address space to be monitored. If the last character of the name is an asterisk (*), the name is to be treated as a generic address-space name prefix; that is, any address space whose name matches the character string up to but not including the asterisk is monitored.								
<i>percent</i>	Is the percentage of total system paging for which an address space must be responsible before an exception message is issued for that address space.								
<i>type</i>	Is the type of address space to be monitored, where <table style="margin-left: 20px;"> <tr> <td>B</td> <td>Indicates a batch job</td> </tr> <tr> <td>S</td> <td>Indicates a started task</td> </tr> <tr> <td>T</td> <td>Indicates a TSO user</td> </tr> <tr> <td>A</td> <td>Indicates any address space matching the specified name; the default</td> </tr> </table>	B	Indicates a batch job	S	Indicates a started task	T	Indicates a TSO user	A	Indicates any address space matching the specified name; the default
B	Indicates a batch job								
S	Indicates a started task								
T	Indicates a TSO user								
A	Indicates any address space matching the specified name; the default								
<i>paging rate</i>	Is the paging rate-per-second that must be reached before exception messages are issued. The default is a system paging rate greater than 25 pages per second.								

Example

Assuming a report interval of 30 seconds, this control statement:

```
apag 4, *, 35, a, 50
```

displays warning messages at 2-minute intervals if the system-paging rate exceeds 50 pages-per-second over the preceding 2 minutes. A warning is issued for any address space doing more than 35 percent of the paging.

Warning Messages

The following warning messages are issued by the APAG sampler:

```
PWSAPAGO *WARNING* TOTAL SYSTEM PAGING RATE IS xxx PAGES/SEC
```

Message PWSAPAGO indicates that the system-paging rate has exceeded the warning threshold.

```
PWSAPAG1 *WARNING* PAGING RATE FOR ttt nnnnnnnn IS xxx% OF SYSTEM TOTAL
```

Message PWSAPAG1 indicates that the specified address space has exceeded the warning threshold for percentage of system paging;

where

ttt Is STC, TSU, or JOB.

nnnnnnnn Is the address space name.

xxx Is the percentage of the total system paging rate.

Message PWSAPGR1 indicates that the specified address space has exceeded the warning threshold for percentage of system paging;

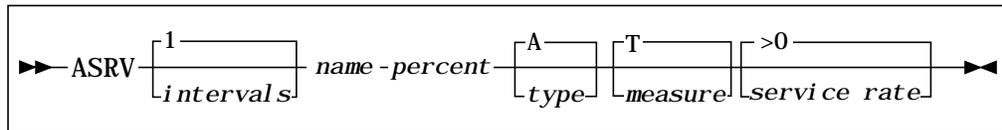
where

<i>ttt</i>	Is STC, TSU, or JOB.
<i>nnnnnnnn</i>	Is the address space name.
<i>ppp</i>	Is the number of pages per second.

ASRV

The Address Space Service Rate Percentage (ASRV) sampler sends a warning message to the operator when an address space uses more than the specified percentage of one or more SRM service components resources during a reporting interval. The message can be suppressed if a given total system service rate threshold is not reached during the reporting interval.

Syntax



where

<i>intervals</i>	Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).								
<i>name</i>	Is the name of the address space to be monitored. If the last character of the name is an asterisk (*), the name is to be treated as a generic address-space name prefix; that is, any address space whose name matches the character string up to but not including the asterisk is monitored.								
<i>percent</i>	Is the percentage of the total system service rate that an address space must be responsible for before an exception message is issued for that address space.								
<i>type</i>	Is the type of address space to be monitored, where <table> <tr> <td>B</td> <td>Indicates a batch job</td> </tr> <tr> <td>S</td> <td>Indicates a started task</td> </tr> <tr> <td>T</td> <td>Indicates a TSO user</td> </tr> <tr> <td>A</td> <td>Indicates any address space matching the specified name; the default</td> </tr> </table>	B	Indicates a batch job	S	Indicates a started task	T	Indicates a TSO user	A	Indicates any address space matching the specified name; the default
B	Indicates a batch job								
S	Indicates a started task								
T	Indicates a TSO user								
A	Indicates any address space matching the specified name; the default								
<i>measure</i>	Specifies the SRM component use rate to be monitored: <table> <tr> <td>C</td> <td>For CPU service units</td> </tr> <tr> <td>I</td> <td>For I/O service units</td> </tr> <tr> <td>M</td> <td>For MSO service units</td> </tr> <tr> <td>T</td> <td>For total service units; the default</td> </tr> </table>	C	For CPU service units	I	For I/O service units	M	For MSO service units	T	For total service units; the default
C	For CPU service units								
I	For I/O service units								
M	For MSO service units								
T	For total service units; the default								
<i>service rate</i>	Is the system SRM-resource service rate (in service units per second) that must be reached before exception messages are issued. The default is a system service rate greater than zero SUs per second.								

Example

Assuming a report interval of 30 seconds, this control statement:

```
asrv 4, *, 35, a, t, 50
```

displays warning messages at 2-minute intervals whenever an address space being monitored uses more than the 35 percent of total SRM service components during a given reporting interval. You can suppress a message if a given total system service rate threshold of 50 is not reached during a reporting interval.

Warning Messages

The following warning messages are issued by the ASRV sampler:

```
PWSASRV0 WARNING* ADDR SPACE mmm SERVICE RATE IS sssss SU/SEC
```

Message PWSASRV0 indicates that the system service-rate has exceeded the warning threshold;

where

mmm Is the SRM service component being monitored.

sssss Is the number of service units.

```
PWSASRV1 WARNING* mmm SERVICE RATE FOR ttt jjjjjjjj IS ppp% OF SYSTEM TOTAL
```

Message PWSASRV1 indicates that the specified address space has exceeded the warning threshold for percentage of system SRM component utilization;

where

mmm Is the SRM service component being monitored.

ttt Is STC, TSU, or JOB.

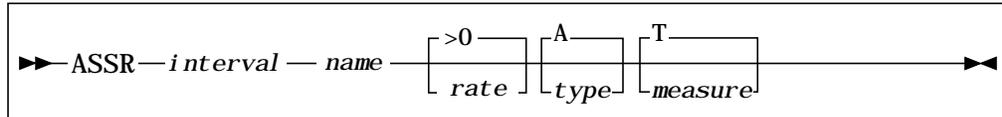
jjjjjjjj Is the address space name.

ppp Is the address space SRM component used as a percentage of system total.

ASSR

The Address Space Service Unit Rate (ASSR) sampler sends a warning message to the operator when an address space uses SRM service components at a rate that exceeds the defined threshold during a reporting interval. The message can be suppressed if a given total system-service rate threshold is not reached during a reporting interval.

Syntax



where

<i>interval</i>	Is the number of report intervals between warning messages.								
<i>name</i>	Is the name of the address space to be monitored. If the last character of the name is an asterisk (*), the name is to be treated as a generic address-space name prefix; that is, any address space whose name matches the character string up to but not including the asterisk is monitored.								
<i>rate</i>	Is the SRM component service rate for an address space that must be reached before exception messages are issued. The default is a system service rate greater than zero SUs per second.								
<i>type</i>	Is the type of address space to be monitored, where <table> <tr> <td>B</td> <td>Indicates a batch job</td> </tr> <tr> <td>S</td> <td>Indicates a started task</td> </tr> <tr> <td>T</td> <td>Indicates a TSO user</td> </tr> <tr> <td>A</td> <td>Indicates any address space matching the specified name; the default</td> </tr> </table>	B	Indicates a batch job	S	Indicates a started task	T	Indicates a TSO user	A	Indicates any address space matching the specified name; the default
B	Indicates a batch job								
S	Indicates a started task								
T	Indicates a TSO user								
A	Indicates any address space matching the specified name; the default								
<i>measure</i>	Specifies the SRM service measure to be monitored as one of the following: <table> <tr> <td>C</td> <td>For CPU service</td> </tr> <tr> <td>I</td> <td>For I/O service</td> </tr> <tr> <td>M</td> <td>For MSO service</td> </tr> <tr> <td>T</td> <td>For total service; the default</td> </tr> </table>	C	For CPU service	I	For I/O service	M	For MSO service	T	For total service; the default
C	For CPU service								
I	For I/O service								
M	For MSO service								
T	For total service; the default								

Example

Assuming an interval of 30 seconds, this control statement:

```
assr 4, *, 100, a, t
```

displays warning messages at 2-minute intervals whenever any address space uses more than a total of 100 SRM service units during a given reporting interval.

Warning Message

The following warning message is issued by the ASSR sampler:

```
PWSASSR1 WARNING* mmm SERVICE RATE FOR ttt jjjjjjjj IS rrrrr SU/SEC
```

Message PWSASSR1 indicates that the address-space service unit rate has exceeded the warning threshold;

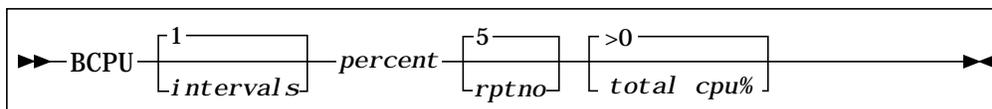
where

<i>mmm</i>	Is the SRM service component being monitored.
<i>ttt</i>	Is STC, TSU, or JOB.
<i>jjjjjjjj</i>	Is the address space name.
<i>rrrrr</i>	Is the number of service units.

BCPU

The Batch CPU (BCPU) sampler sends a warning message to the operator when a batch address space uses more than a specified percentage of the total CPU time during a reporting interval. The message can be suppressed if a given threshold for total CPU usage is not reached during the reporting interval.

Syntax



where

<i>intervals</i>	Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).
<i>percent</i>	Is the percentage of total system CPU time that a batch address space must be responsible for before an exception message is issued for that address space.
<i>rptno</i>	Is the maximum number of address spaces to be listed whenever a CPU overload condition is detected. The default is 5.
<i>total cpu %</i>	Is the CPU busy percentage that must be reached before exception messages are issued. The default is a system CPU busy percentage greater than zero.

Example

Assuming a report interval of 30 seconds, this control statement:

```
bcpu 4, 35, 5, 50
```

displays up to 5 warning messages at 2-minute intervals if the system-CPU busy percentage exceeds 50 percent over the preceding 2 minutes. A warning is issued for any batch address space that has been active 35 percent of the total CPU busy time.

Warning Message

The following warning message is issued by the BCPU sampler:

```
PWSBCPU1 WARNING* CPU USAGE IS ppp% FOR JOBjjjjjjj
```

Message PWSBCPU1 indicates that the specified address space has exceeded the warning threshold for percentage of system-CPU busy time;

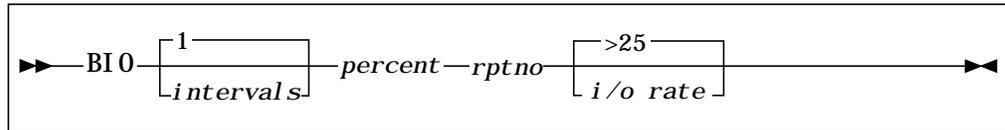
where

<i>ppp</i>	Is the address space busy percentage.
<i>jjjjjjj</i>	Is the address space name.

BIO

The Batch I/O (BIO) sampler sends a warning message to the operator when a batch address space uses more than a specified percentage of the total system I/O activity during a reporting interval. The message can be suppressed if a given total system I/O rate threshold is not reached during a reporting interval.

Syntax



where

- intervals* Specifies the length in intervals of the sample period. The default sample period is 30 seconds (1 interval).
- percent* Is the percentage of total system I/O that a batch address space must be responsible for before an exception message is issued for that address space.
- rptno* Is the maximum number of batch address spaces to be listed whenever a batch I/O overload condition is detected.
- i/o rate* Is the number of EXCPs-per-second that must be reached before exception messages are issued. The default is a system I/O rate greater than 25 EXCPs per second.

Example

Assuming a report interval of 30 seconds, this control statement:

```
bio 4, 25, 5, 50
```

displays warning messages at 2-minute intervals if the system I/O rate exceeds 50 EXCPs-per-second over the preceding 2 minutes. A warning is issued for any batch address space doing more than 25 percent of the I/O.

Warning Messages

The following warning messages are issued by the BIO sampler:

```
PWSBI 000 WARNING* TOTAL SYSTEM I/O RATE IS xxx EXCPS/SEC
```

Message PWSBIO00 indicates that the system-I/O rate has exceeded the warning threshold.

```
PWSBI 001 WARNING* I/O RATE FOR JOB nnnnnnnn IS xxx% OF SYSTEM TOTAL
```

Message PWSBIO01 indicates that the specified address space has exceeded the warning threshold for percentage of system I/O;

where

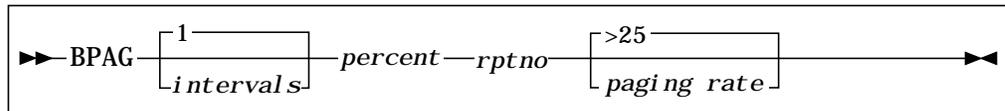
nnnnnnnn Is the address space name.

xxx Is the percent of the total system I/O rate.

BPAG

The Batch Paging (BPAG) sampler sends a warning message to the operator when a batch address space uses more than a specified percentage of the total system paging activity during a reporting interval. The message can be suppressed if a given threshold for the total system paging rate is not reached during the reporting interval.

Syntax



where

- intervals* Specifies the length in intervals of the sample period. The default sample period is 30 seconds (1 interval).
- percent* Is the percentage of total system paging that a batch address space must be responsible for before an exception message is issued for that address space.
- rptno* Is the maximum number of batch address spaces to be listed whenever a batch paging overload condition is detected.
- paging rate* Is the paging rate-per-second that must be reached before exception messages are issued. The default is a system paging rate greater than 25 pages per second.

Example

Assuming a report interval of 30 seconds, this control statement:

```
bpag 4, 25, 5, 50
```

displays warning messages at 2-minute intervals if the system paging rate exceeds 50 pages-per-second over the preceding 2 minutes. A warning is issued for any batch address space doing more than 25 percent of the paging.

Warning Messages

The following warning messages are issued by the BPAG sampler:

```
PWSBPAGO WARNING* TOTAL SYSTEM-PAGING RATE IS NOW ppp PAGES/SEC
```

Message PWSBPAG0 indicates that the system paging rate has exceeded the warning threshold.

```
PWSBPAG1 WARNING* PAGING RATE FOR JOB nnnnnnnn IS xxx% OF TOTAL SYSTEM PAGING
```

Message PWSBPAG1 indicates that the specified address space has exceeded the warning threshold for percentage of system paging;

where

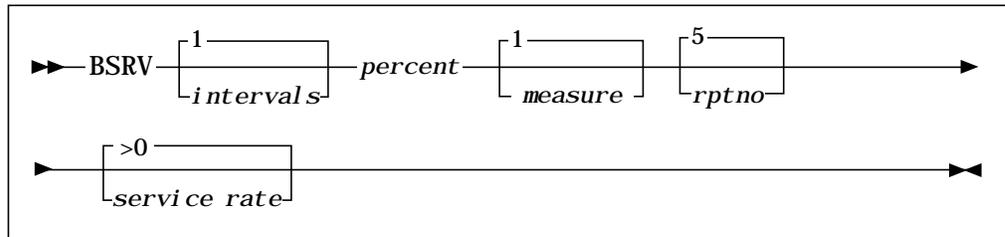
nnnnnnnn Is the address space name.

xxx Is the percent of total system paging rate.

BSRV

The Batch Service (BSRV) sampler sends a warning message to the operator when a batch address space uses more than the specified percentage of one or more SRM service measures during a reporting interval. You can suppress the message if a given total system service-rate threshold is not reached during the reporting interval.

Syntax



where

- intervals* Specifies the length in intervals of the sample period. The default sample period is 30 seconds (1 interval).
- percent* Is the percentage of total system SRM component utilization for which an address space is responsible. This total must be exceeded before an exception message is issued for that address space.
- measure* Specifies the SRM service measure to be monitored as one of the following:
 - C For CPU service units
 - I For I/O service units
 - M For MSO service units
 - T For total service units; the default
- rptno* Is the maximum number of address spaces to be listed whenever a batch service overload condition is detected. The default is 5.
- service rate* Is the system SRM service-consumption rate that must be reached before exception messages are issued. The default is a system service rate greater than zero SUs per second.

Example

Assuming a report interval of 30 seconds, this control statement:

```
bsrv 4, 35, t, 5, 50
```

displays up to 5 warning messages at 2-minute intervals whenever an address space being monitored uses more than the 35 percent of total SRM components service during a given reporting interval. You can suppress the message if a given total system service rate threshold of 50 is not reached during a reporting interval.

Warning Messages

The following warning messages are issued by the BSRV sampler:

```
PWSBSRV0 WARNING* ADDR SPACE mmm SERVICE RATE IS sssss SU/SEC
```

Message PWSBSRV0 indicates that the system paging rate has exceeded the warning threshold;

where

mmm Is the SRM service component being monitored.

sssss Is the number of service units per second.

```
PWSBSRV1 WARNING* mmm SERVICE RATE FOR JOB jjjjjjj IS ppp% OF SYSTEM TOTAL
```

Message PWSBSRV1 indicates that the specified address space has exceeded the warning threshold for percentage of system SRM component utilization;

where

mmm Is the SRM service component being monitored.

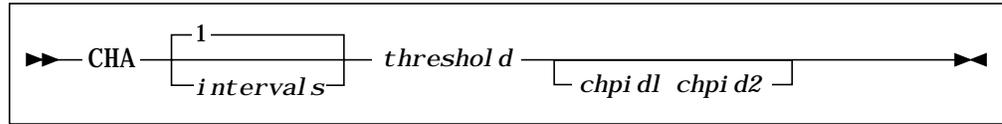
jjjjjjj Is the address space name.

ppp Is the address space SRM component use as percentage of system total.

CHA

The Channel (CHA) sampler monitors channel-path usage and sends a warning message to the operator when a channel path exceeds the defined utilization-percentage threshold.

Syntax



where

interval s Specifies the length in intervals of the sample period. The default sample period is 30 seconds (1 interval).

threshold Is the average utilization high threshold (0-100 percent).

chpi d1 Is the lowest path number in the range of channel paths to be monitored; must be 2 digits.

chpi d2 Is the highest path number in the range of channel paths to be monitored; must be 2 digits.

Note: Parameters *chpi d1* and *chpi d2* are optional. If they are omitted, all channel paths are monitored.

Example 1

Assuming a report interval of 30 seconds, this control statement:

```
cha 2, 35, 01, 10
```

displays warning messages at 1-minute intervals if any channel path in the range 01 through 10 exceeds 35 percent utilization.

Example 2

Assuming a report interval of 30 seconds, this control statement:

```
cha 1, 40
```

displays warning messages at 30-second intervals if any channel path utilization exceeds 40 percent.

Warning Message

The following warning message is issued by the CHA sampler:

```
PWSCHA01 WARNING* CHANNEL xx ON CPU y UTILIZATION IS zz%
```

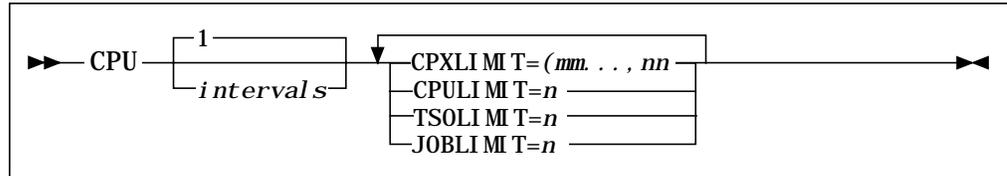
where

<i>xx</i>	Is the physical channel number (0 to 15).
<i>y</i>	Is the CPU number (0 or 1).
<i>zz</i>	Is the percent utilization.

CPU

The CPU Utilization (CPU) sampler monitors CPU usage by system, by TSO, and by individual jobs.

Syntax



where

- i n t e r v a l s* Specifies the length in intervals of the sample period. The default sample period is 30 seconds (1 interval).
- CPXLI MI T=nn** Provides the ability to establish warning thresholds (in percent) for each CPU in a multiprocessor system. A warning message will be issued when the average CPU utilization for the specified CPU(s) exceeds the threshold value(s) during the interval. Valid values are 0-99. A value of zero (the default) indicates that no warning should be issued. Values are positional and correspond to CPU0 through CPU15 inclusive. Multiple values must be enclosed in parenthesis and separated by commas.
- CPULI MI T=nn** Is the average CPU utilization for all CPUs in a multiprocessor system that causes a warning message to be issued. Specify 2 digits.
- TSOLI MI T=nn** Is the percent CPU utilization by TSO users that causes a warning message to be issued. Specify 2 digits.
- JOB LI MI T=nn** Is the percent CPU utilization by individual jobs, TSO users, or started tasks that causes a warning message to be issued. Specify 2 digits.

Note: If a threshold specification is omitted or is zero, no warning messages are produced.

Example

Assuming a report interval of 30 seconds, this control statement:

```
cpu 10, cpxl i m i t=99, tsol i m i t=50, job l i m i t=20
```

displays warning messages at 5-minute intervals if CPU0 usage exceeds 99 percent, TSO usage exceeds 50 percent, or any job uses more than 20 percent of the CPU.

Warning Messages

The following warning messages are issued by the CPU sampler:

```
PWSCPU00 WARNING* CPUx USAGE IS xxx%
```

Message PWSCPU00 indicates that the utilization of CPU_x exceeds the warning threshold.

```
PWSCPU01 WARNING* CPU COMPLEX USAGE IS xxx
```

Message PWSCPU01 indicates that the average CPU usage for all CPUs in the complex exceeds the warning threshold.

```
PWSCPU02 WARNING* CPU USAGE IS xxx% FOR TSO
```

Message PWSCPU02 indicates that the total CPU usage by TSO users exceeds the warning threshold.

```
PWSCPU03 WARNING* CPU USAGE IS nn% FOR aaa xxxxxxxx
```

Message PWSCPU03 indicates that the CPU usage of address space name xxxxxxxx exceeds the warning threshold. The value aaa can be JOB, TSU, or STC.

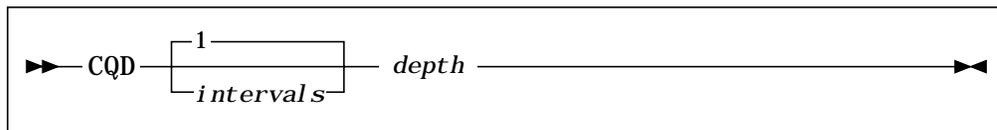
```
PWSCPU90 PARAMETER xxxxxxxx INVALID FOR CPU SERVICE
```

Message PWSCPU90 indicates that parameter xxxxxxxx contains invalid data.

CQD

The CPU Queue Depth (CQD) sampler monitors the CPU active task queue. It also sends a warning message to the operator when the average queue depth of active tasks (both those in memory and those ready for dispatching) exceeds a threshold you specify.

Syntax



where

- interval s* Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).
- depth* Is the threshold count of active tasks waiting.

Example

Assuming a report interval of 30 seconds, this control statement:

```
cqd 1, 10
```

displays warning messages at 30-second intervals if the average CPU queue depth exceeds 10.

Warning Message

The following warning message is issued by the CQD sampler:

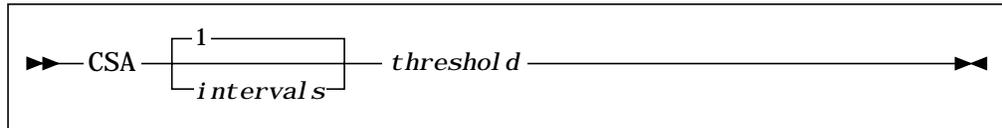
```
PWSCQD01 WARNING* CPU QUEUE DEPTH IS xxx
```

Message PWSCQD01 indicates that *xxx* is the average number of active tasks waiting.

CSA

The CSA and SQA Utilization (CSA) sampler monitors CSA usage and system-defined critical thresholds for SQA usage, and anticipates problems caused by CSA or SQA usage that exceeds the thresholds.

Syntax



where

intervals Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).

threshold Is the percent utilization of CSA that causes a warning message to be issued.

Note: For SQA usage, bits SQATHRS1 and SQATHRS2 in the GDA are monitored.

Example

Assuming a report interval of 30 seconds, this control statement:

```
csa 1, 80
```

displays a warning message at 30-second intervals whenever CSA utilization exceeds 80 percent.

Warning Messages

The following warning messages are issued by the CSA sampler:

```
PWSCSA01 WARNING* CSA|ECSA USAGE IS xxx%; yyyyK ARE FREE
```

Message PWSCSA01 indicates that CSA or ECSA usage exceeds the specified threshold; yyyy represents the number of kilobytes of CSA still available.

```
PWSCSA02 WARNING* SQA CRITICAL THRESHOLD EXCEEDED
```

Message PWSCSA02 indicates that SQA utilization has reached a critical level.

```
PWSCSA03 WARNING* SQA APPROACHING CRITICAL LEVEL
```

Message PWSCSA03 indicates that SQA utilization is reaching a critical level.

PWSCSA90 SPECIFIED THRESHOLD GREATER THAN 100%
--

Message PWSCSA90 indicates that the CSA control statement was rejected because the specified threshold is invalid.

CSMJ

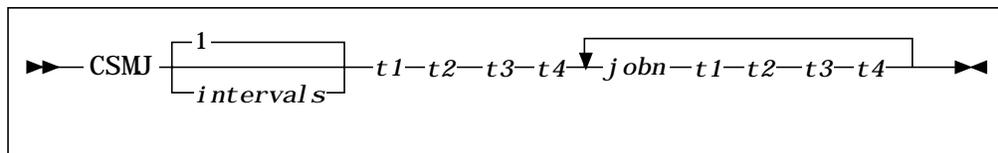
The CSMJ sampler monitors the percentage of common storage use by jobnames and sends a warning message to the operator for each job that meets or exceeds the specified thresholds. This allows the operator take action before a common storage shortage adversely affects the entire system. The thresholds represent the percent of total available common storage (CSA, ECSA, SQA, or ESQA) that has been allocated by any single jobname.

Global thresholds apply to all jobs except *MASTER* and *SYSTEM*. The default global thresholds are 3% for CSA and SQA, and 2% for ECSA and ESQA.

The master address space (*MASTER*) and storage attributed to the system (SYSTEM*) will only be checked when specified limits are provided for these names.

You can specify threshold overrides for specific jobnames. Address spaces that can allocate large amounts of common storage (such as JES2, IMS, and DB2) should have their jobnames specified and their threshold values raised accordingly, which prevents the issuance of multiple warning messages.

Syntax



where

<i>intervals</i>	Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).
<i>t1</i>	Is the percent allocation of CSA by any jobname that causes a warning message to be issued.
<i>t2</i>	Is the percent allocation of ECSA by any jobname that causes a warning message to be issued.
<i>t3</i>	Is the percent allocation of SQA by any jobname that causes a warning message to be issued.
<i>t4</i>	Is the percent allocation of ESQA by any jobname that causes a warning message to be issued.
<i>jobn</i>	Is a jobname followed by 4 threshold values that apply only to that specific jobname; up to 32 job or threshold values can be specified.

Note: To monitor all jobs, use *SYSTEM* as the jobname.

Example

Assuming a report interval of 30 seconds, this control statement:

```
csmj 3, 2, 3, 2,
    *master*, 4, 3, 4, 3
```

displays a warning message at 30-second intervals whenever any job has allocated more than the global thresholds, or for jobname *MASTER*, whenever it has allocated more than 4 percent of CSA or SQA and more than 3 percent of ECSA or ESQA.

Note: Control statements ending with a comma (,) indicate the statement is continued on the next input line. Continuation lines must be indented if the jobname has an asterisk as the first character.

Warning Messages

The following warning messages are issued by the CSMJ sampler:

```
PWSCSM01 *WARNING* xxxxxxxx yyyy HAS ALLOCATED nn% OF zzzz
```

Message PWSCSM01 indicates that jobname *xxxxxxx* in ASID *yyyy* has allocated *nn* percent of available common storage. The common storage area (CSA, ECSA, SQA, or ESQA) is indicated by *zzzz*.

```
PWSCSM02 *WARNING* FAILURE EXTRACTING CS MONITOR DATA
```

Message PWSCSM02 indicates that an error has occurred in the COMMON STORAGE MONITOR component of BBXS. Contact Boole & Babbage Customer Support.

```
PWSCSM03 *WARNING* CS MONITOR IS NOT ACTIVE
```

Message PWSCSM03 indicates that COMMON STORAGE MONITOR must be active to use the CSMJ sampler. Refer to the *COMMON STORAGE MONITOR User Guide* for information about starting the COMMON STORAGE MONITOR.

```
PWSCSM04 *WARNING* BBXS LEVEL DOES NOT SUPPORT CS MONITOR
```

Message PWSCSM04 indicates that the active BBX subsystem is not at the required level to support COMMON STORAGE MONITOR. Ensure that current maintenance has been applied to FMID BBBB16. An IPL or execution of the BBXSINIT batch job is required to activate the new level of BBXS after applying maintenance.

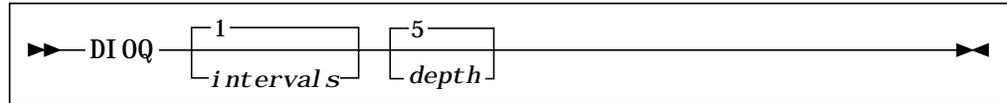
```
PWSCSM05 *WARNING* BBXS IS NOT ACTIVE
```

Message PWSCSM05 indicates that BBXS must be active to use the CSMJ sampler. BBXS is started automatically when MainView for OS/390 is activated. Contact Boole & Babbage Customer Support.

DIOQ

The DASD I/O Queue (DIOQ) sampler displays information for a DASD device when the number of I/Os queued to the device equal or exceed the specified queue depth.

Syntax



where

intervals Is the number of intervals between report periods; the default report period is 30 seconds (1 interval).

depth **Is the queue depth necessary for the device to be reported. For example, if 3 is specified, all DASD devices with 3 or more queued requests are reported. The minimum value is 1 and the maximum is 999; the default is 5.**

Example

Assuming a report interval of 30 seconds, this control statement:

```
di oq 4, 3
```

displays warning messages at 2-minute intervals if the I/O Queue depth is greater than or equal to 3.

Warning Message

The following message is issued by the DIOQ sampler:

```
PWSDIOQ0 *WARNING* Q=qqq nnnn aaa volser s mmm jobname PENDING
```

Message PWSDIOQ0 indicates that the I/O Queue depth has exceeded the warning threshold;

where

qqq Is the I/O queue depth for the DASD device.

nnnn Is the device type.

aaa Is the device address.

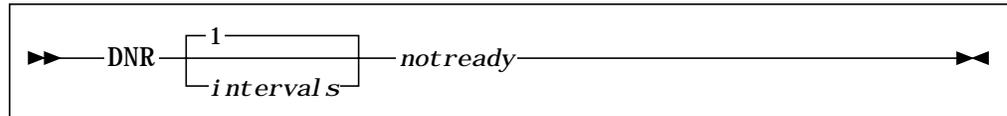
volser Is the volume serial number.

<i>s</i>	Is the special status indicator. The letter P indicates that the device contains an active paging data set. The letter R indicates that the device is currently reserved.
<i>mmm</i>	Is the mount attribute for the volume.
<i>jobname</i>	Is the jobname for the job with I/O in progress at the time the device was checked.
PENDING	If present, indicates that I/O has been started for the device but has not completed. This can indicate a problem, such as the device is reserved by another system. However, this is a normal condition for paging and other applications that use command chaining to continue a single I/O.

DNR

The Device Not Ready (DNR) sampler monitors tape and DASD devices online but not ready and notifies the operator when online devices have been not ready for more than a specified interval.

Syntax



where

- intervals* Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).
- notready* Is the number of minutes a device must remain not ready for a warning message to be issued.

Example

Assuming a report interval of 30 seconds, this control statement:

```

dnr 4, 2
    
```

displays warning messages at 2-minute intervals for each device that has been online but not ready for over 2 minutes.

Warning Message

The following warning message is issued by the DNR sampler:

```

PWSDNR01 WARNING* DEVICE nnn (vvvvvv) NOT READY FOR xx MIN
    
```

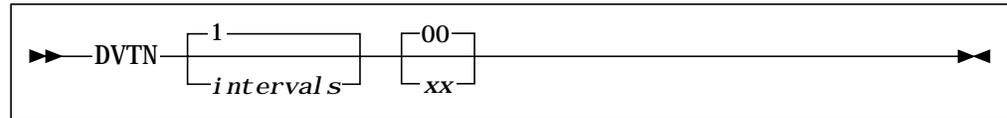
Message PWSDNR01 indicates that device *nnn* has been online but not ready for a period of time that exceeds the warning threshold.

Note: A blank volume serial number appears when volume rollover occurs.

DVTN

The DVTN sampler displays deviation from a list of jobs that must be active during specified time intervals and notifies you when a job that should be active is not. This list of jobs resides in a CONFIG xx member of SYS1.PARMLIB.

Syntax



where

<i>intervals</i>	Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).
00	Is the default suffix of the CONFIG xx member.
xx	Is the suffix of the CONFIG xx member in SYS1.PARMLIB that is to be used.

The entries in the CONFIG xx member must begin in column 1. The format of the entries is

```
*/NAME=jobname TIME=hhmm-hhmm ACT=command
```

where

<i>jobname</i>	Is the jobname (or started task ID for started tasks).
<i>hhmm-hhmm</i>	Are the beginning and ending times for the interval that the specified job must be active. A beginning value of four zeros (0000- <i>hhmm</i>) indicates that the job should ALWAYS be active. <ul style="list-style-type: none"> Valid values for <i>hh</i> are 00 to 23 Valid values for <i>mm</i> are 00 to 59
<i>command</i>	Is an optional field that you can substitute with any OS/390 command. The command is executed if the targeted job is not active during the specified time interval. When ACT is specified, <i>command</i> cannot be blank.

Example

Assuming a report interval of 30 seconds, this control statement:

```
dvtn 10,00
```

displays a warning message at 5-minute intervals if the jobs in CONFIG00 are not executing.

Warning Message

The following warning message is issued by the DVTN sampler:

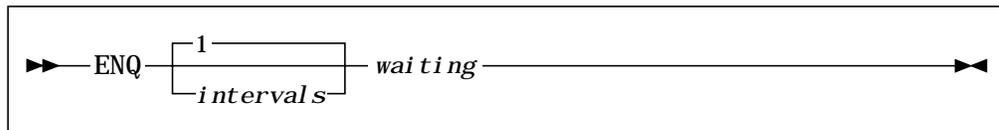
```
PWSDV8I  WARNING as_name SHOULD BE ACTIVE hhmm-hhmm
```

Message PWSDV8I indicates that the listed address space is not active during the indicated shift. You should correct the operational status of the address space.

ENQ

The Enqueue Conflicts (ENQ) sampler monitors enqueue conflicts and notifies the operator when jobs have been waiting for enqueued resources for more than a specified interval.

Syntax



where

i n t e r v a l s Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).

w a i t i n g Is the number of minutes that an enqueue conflict must exist for a warning message to be issued.

Example

Assuming a report interval of 30 seconds, this control statement:

```
enq 4, 5
```

displays warning messages at 2-minute intervals if any enqueue conflicts have existed for over 5 minutes.

Warning Message

The following warning message is issued by the ENQ sampler:

```
PWSENQ01 WARNING* ENQUEUE CONFLICT(S) PRESENT OVER xxx MIN
```

Message PWSENQ01 indicates that one or more jobs have been waiting for resources for longer than the specified threshold. To display the enqueue situation, use service ENQUEUEES.

INT

The Interval (INT) service changes the reporting base interval and the DIE sample interval. For each sampler, you can specify a multiple of the report base interval in the interval parameter of the control statement.

Syntax

```
▶▶—INT—report interval— sample interval —◀◀
```

where

report interval Is the interval between report phases in seconds. The range is from 5 to 60 seconds; the default is 30 seconds.

sample interval Is the interval between DIE samples in milliseconds. The range is from 100 to 1000 milliseconds; the default is CPU-model dependent:

263ms 3081, 3084

511ms 3083, 4381, all others

Example

To set a report interval of 60 seconds and a sample interval of 1000 milliseconds, place this control statement in BBPARM library member PWSCPMxx:

```
int 60, 1000
```

The base interval between report phases is 60 seconds. The system is sampled under the DIE once every 1000 milliseconds (1 second).


```
PWSJET02 WARNING* CLASS n JOB nnnnnn jjjjjjj EXECUTING xxxxx MINUTES
```

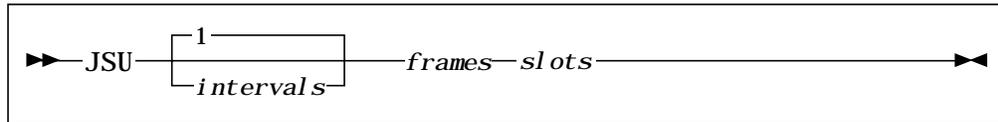
where

<i>n</i>	Is the job class.
<i>nnnnnn</i>	Is the JES2 job number.
<i>jjjjjjj</i>	Is the jobname.
<i>xxx</i>	Is the elapsed time in minutes.

JSU

The Job Storage Usage—Real & Auxiliary (JSU) sampler monitors all jobs' usage of ASM slots and real frames, and identifies jobs that are overusing real storage and allocated slots of auxiliary storage.

Syntax



where

<i>interval s</i>	Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval). You can specify up to 99.
<i>frames</i>	Is the maximum number of frames that can be held by a job (real storage) before a message is issued. You can specify up to 99999.
<i>slots</i>	Is the maximum number of slots that can be held by a single job (auxiliary storage) before a message is issued. You can specify up to 99999.

Note: If a threshold specification is omitted or is zero, no warning message is issued.

Examples

Assuming a report interval of 30 seconds, this control statement:

```
jsu 10, 100, 200
```

displays warning messages every 5 minutes for each job that holds more than:

- 200 slots
- 100 frames

Warning Messages

The following warning messages are issued by the JSU sampler:

```
PWSJSU01 WARNING* JOBjjjjjjj HOLDS xxx FRAMES
```

where

<i>jjjjjjj</i>	Is the jobname.
<i>xxx</i>	Is the count of frames held.

```
PWSJSU02 WARNING* JOBjjjjjjj HOLDS yyy SLOTS
```

where

jjjjjjj

Is the jobname.

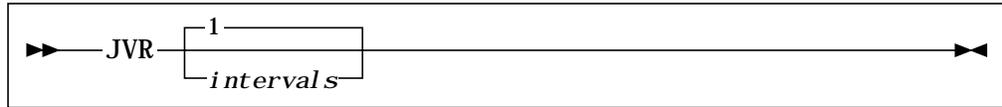
yyy

Is the count of slots held.

JVR

The Job Awaiting V=R Region (JVR) sampler monitors jobs awaiting a V=R region and notifies the operator if any job is waiting for a V=R region.

Syntax



where

i n t e r v a l s Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).

Example

Assuming a report interval of 30 seconds, this control statement:

```
jvr 10
```

displays warning messages at 5-minute intervals for each job waiting for a V=R region.

Warning Messages

The following warning messages are issued by the JVR sampler:

```
PWSJVR01 WARNING* JOBjjjjjjjj WAITING FOR V=R REGION
```

where

jjjjjjjj Is the jobname of the waiting job.

```
PWSLCQ01 WARNING* I/O QUEUE FOR LOGICAL CHANNELyyyy ISxxx
```

where

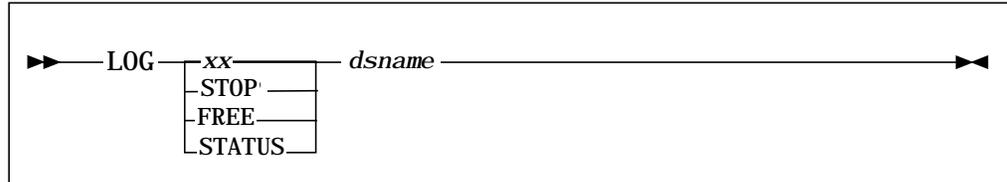
xxx Is the average queue depth.

yyyy Is the physical channels on this logical channel queue.

LOG

The LOG service controls the logging system. The logging system provides the ability to periodically execute a set of services with the output directed to a data set that you specified.

Syntax



where

- xx** Is the suffix of the \$\$INLGxx member used to start logging. The \$\$INLGxx member contains control statements that define the services to be executed and control the logging process. This parameter also implies start.
- dsname** Is the name of the partitioned data set that contains the \$\$INLGxx member. If this parameter is omitted, the name you last specified is used. If you did not previously specify a *dsname*, the data set allocated to DDNAME BBPARM is used.
- STOP** Indicates that logging is to be stopped.
- FREE** Indicates that the logging output data set is to be freed. FREE implies stop.
- STATUS** Indicates that the status of the logging system is to be displayed.

\$\$INLGxx Control Statements

The verb (statement type) must begin in column 1. One or more blanks are allowed between the verb and the operand. Operands cannot extend beyond column 72.

- CLASS c** Indicates that the logging output is to be written to a SYSOUT data set of the specified class. CLASS is mutually exclusive with the DSN statement.
- CMD Command** It is a command that is to be periodically executed. Use multiple CMD statements for multiple commands. Commands are executed in the order that they appear.
- CNT n** Specifies that logging is to be terminated when *n* number of cycles have been executed. Also, see the STOP statement.
- DSN dsname** Specifies that logging output is to be written to the preallocated data set *dsname*. If no Class or DSN statement is provided, and a data set has been allocated to DDNAME LOG, the output will be written to that data set (which is not supported under *MV Manager for MVS*). Otherwise, it will be written to SYSOUT=A.

EJC	Indicates the output produced by the command that follows starts on a new page. The EJC statement should be followed by one or more CMD statements.
INCR <i>n</i>	Indicates the number (<i>n</i>) of seconds (increment) in an interval. The default is 30 seconds. The product of INCR and INT determines the number of seconds between logging cycles.
INT <i>n</i>	Indicates the number (<i>n</i>) of intervals in a logging cycle. The default is one interval. The product of INCR and INT determines the number of seconds between logging cycles.
START <i>hhmm</i>	Indicates the time (<i>hhmm</i>) that logging is to start. Specify 1300 for 1:00pm. Logging starts immediately if you do not provide a START statement.
STOP <i>hhmm</i>	Indicates the time (<i>hhmm</i>) that logging is to terminate. Specify 1200 for 12:00pm (noon) and 2400 for 12:00am (midnight). Note: If you provide both CNT and STOP statements, logging will terminate when the first condition is reached.
HOLD	Indicates that the SYSOUT output is to be held. If HOLD and NOHOLD are not specified, the default for the SYSOUT class will prevail. The HOLD statement is ignored if logging output is not written to a SYSOUT data set.
NOHOLD	Indicates that the SYSOUT output is <i>not</i> to be held. If HOLD and NOHOLD are not specified, the default for the SYSOUT class will prevail. The NOHOLD statement is ignored if logging output is not written to a SYSOUT data set.
MAXOUT <i>n</i>	Indicates the maximum number (<i>n</i>) of lines that will be written to a SYSOUT data set. Logging will be terminated when <i>n</i> lines have been written to SYSOUT. The default is to write an infinite number of lines. MAXOUT is only applicable to SYSOUT data sets.

Examples

Member \$\$INLG01

```
INCR 60
INT 5
DSN VAM3.LOG
START 2200
STOP 2259
CMD CSA, MAP
```

Member \$\$INLG01 will cause the command CSA, MAP to be executed once every five minutes between 10:00pm and 10:59pm. The output will be written to data set VAM3.LOG.

Member \$\$INLG02

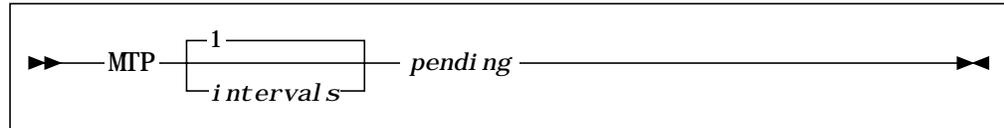
```
INCR 30  
INT 2  
CNT 60  
HOLD  
CLASS A  
CMD MTP  
CMD ENQ  
CMD CPU 5
```

Member \$\$INLG02 causes the commands MTP, ENQ, and CPU 5 to be executed once a minute for a total of 60 times. The output will be written to SYSOUT Class A and placed on the HOLD queue.

MTP

The Mounts Pending (MTP) sampler monitors tape and DASD-mount requests, and notifies the operator when outstanding mount requests have been pending for more than a specified interval.

Syntax



where

intervals Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).

pending Is the number of minutes a mount must remain pending for a warning message to be issued.

Example

Assuming a report interval of 30 seconds, this control statement:

```
mtp 4, 10
```

displays warning messages at 2-minute intervals for each device with a mount pending for over 10 minutes.

Warning Message

The following warning message is issued by the MTP sampler:

```
PWSMTP01 WARNING* MOUNT FOR dvn (vvvvvv) OUTSTANDING FOR xx MINUTES
```

Message PWSMTP01 indicates that the mount request for device *dvn* has been outstanding for a period of time that exceeds the warning threshold.

Note: A blank volume serial number appears when volume rollover occurs.

NRQ

The Non Reusable Queue (NRQ) sampler displays a warning message when the following conditions occur:

- The number of available address space vector table slots (ASIDs) reserved for replacing address space IDs marked as non-reusable drops below the specified threshold value (*limit1*), **and**
- The number of slots available for starting new address spaces drops below the specified threshold value (*limit2*).

Syntax

```
▶▶—NRQ—intervals—limit1—limit2—◀◀
```

where

*interval*s Specifies the length of the sample period in intervals. The default interval is 30 seconds.

limit1 Is the threshold for slots reserved for replacing slots marked non-reusable. The number of slots on the non-reusable replacement queue (value in ASVTANR) must be less than or equal to the specified number for a warning message to be issued.

limit2 Is the threshold for the available queue (number of additional address spaces that may be started). The number of slots on the available queue (ASVTAAV) must be less than or equal to the specified number for a warning message to be issued.

Notes: Slots (ASIDs) are marked non-reusable when the job terminates while in a cross-memory environment. They remain unusable until all address spaces that had binds with the address space have ended.

The SYSPROG ASVT service displays the current and original values for the non-reusable replacement queue and the available queue.

Example

To display a warning message for a specified threshold, type:

```
NRQ 2, 5, 10
```

This NRQ sampler checks every 2 intervals to see if there are less than 6 (*limit1* value) slots remaining on the non-reusable replacement queue and less than 11 (*limit2* value) slots on the available queue. A warning message is issued if both *limit1* and *limit2* have been reached. The message is reissued every minute (2 intervals), assuming the default interval value is 30 seconds, unless the number of slots on the available queue exceeds the *limit2* value.

Warning Message

The following warning message is issued by the NRQ service:

```
PWSNRQ01 *WARNING* non-reusable replacement queue length is 4
```

NRQP

The Non Reusable Queue Percentage (NRQP) sampler displays a warning message when the following conditions occur:

- The percentage of available address space vector table slots (ASIDs) reserved for replacing address space IDs marked as non-reusable drops below the specified threshold value (*limit1*), **and**
- The percentage of slots available for starting new address spaces drops below the specified threshold value (*limit2*).

Syntax

```
▶▶ NRQP — interval s — limit 1 — limit 2 —▶▶
```

where

- interval s* Specifies the length of the sample period in intervals. The default interval is 30 seconds.
- limit 1* Is the threshold for slots reserved for replacing slots marked non-reusable. The percentage of slots on the non-reusable replacement queue ($ASVTANR / ASVTNONR * 100$) must be less than or equal to the specified percentage for a warning message to be issued.
- limit 2* Is the threshold for the percentage of address spaces (MAXUSER) that still may be started. The percentage of the original number of slots on the available queue that are still available ($ASVTAAV / ASVTMAXI * 100$) must be less than or equal to the specified percentage for a warning message to be issued.

Notes: Slots (ASIDs) are marked non-reusable when the job terminates while in a cross-memory environment. They remain unusable until all address spaces that had binds with the address space have ended.

The SYSPROG AVST service displays the current and original values for the non-reusable replacement queue and the available queue.

Example

To display a warning message for a specified percentage, type:

```
NRQP 10, 25, 10
```

This NRQP sampler checks every 10 intervals to see if there are less than 26% (*limit1* value) of the slots remaining on the non-reusable replacement queue and less than 10% (*limit2* value) of MAXUSER³ slots on the available queue. A warning message is issued if both *limit1* and *limit2* have been reached. The message is reissued every minute (two intervals), assuming the default interval value is 30 seconds, unless the percentage of slots on the available queue exceeds the *limit2* value.

³ Obtained from ASVTMAXI.

Warning Message

The following warning message is issued by the NRQP service:

```
PWSNRQP1 *WARNING* Non-reusable replacement queue length is 8% (2)
```

OUT

The Jobs Swapped Out (OUT) sampler monitors jobs swapped out of memory for an interval that exceeds the user-specified threshold. It also notifies the operator if any job, TSO user, or started task has been swapped out for an interval that exceeds the threshold. TSO users in input or output terminal wait are not considered.

Syntax



where

- intervals* Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).
- time* Is the threshold for swap-out in minutes.

Example

Assuming a report interval of 30 seconds, this control statement:

```
out 2, 1
```

displays warning messages at 1-minute intervals for jobs, TSO users, or started tasks swapped out longer than 1 minute.

Warning Message

The following warning message is issued by the OUT sampler:

```
PSWOUT01 WARNING* JOB or TSU or STCjjjjjjj SWAPPED OUT FOR xxx MINUTES
```

where

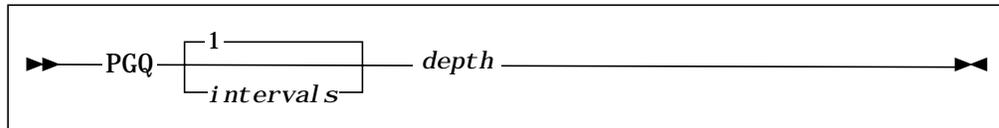
- jjjjjjj* Is the jobname, TSO user, or started task.
- xxx* Is the number of minutes swapped out.

PGQ

The Page Data Set I/O Queue (PGQ) sampler monitors the depth of I/Os queued to page data sets and notifies the operator if the average I/O queue depth to any page data set exceeds a specified threshold.

Note: Average queue depth computation includes samples taken when queue depth is zero; therefore, it can be less than one.

Syntax



where

intervals Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).

depth Is the threshold average of I/Os queued to any page or swap data set.

Example

Assuming a report interval of 30 seconds, this control statement:

```
pgq 4, 5
```

displays warning messages at 2-minute intervals if the average count of I/Os queued to any page-data set exceeds 5.

Warning Message

The following warning message is issued by the PGQ sampler:

```
PWSPGQ01 WARNING* I/O QUEUE DEPTH FOR PAGE DATA SET n ON vvvvvv IS xx
```

where

n Is the data set number.

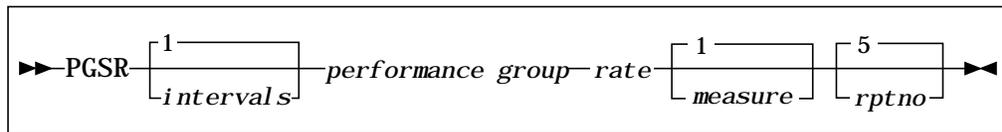
vvvvvv Is the volume serial of device.

xx Is the I/O queue depth.

PGSR

The Performance Group Service (PGSR) sampler sends a warning message to the operator when an address space in a performance group uses an SRM component during a reporting interval at a rate that exceeds the specified threshold. You can suppress the message if a given total system service rate threshold is not reached during a reporting period.

Syntax



where

<i>intervals</i>	Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).
<i>performance group</i>	Is the performance group number.
<i>rate</i>	Is the SRM component-service rate for a performance group that must be reached before exception messages are issued.
<i>measure</i>	Specifies the SRM service measure to be monitored as one of the following: <ul style="list-style-type: none"> C For CPU service units I For I/O service units M For MSO service units T For total service units; the default
<i>rptno</i>	Is the maximum number of address spaces that are listed whenever the specified rate is exceeded. The default is 5.

Example

Assuming a report interval of 30 seconds, this control statement:

```
pgsr 8, 2, 100, t, 5
```

displays up to 5 warning messages at 4-minute intervals when any one address space in the performance group field uses more than a total of 100 SRM service units during a given reporting interval.

Warning Message

The following warning message is issued by the PGSR sampler:

PWSPGSR1 WARNING* *mmm* SERVICE RATE FOR *ttt* *jjjjjjjj* (PG *ggg*) is *ppppp*
SU/SEC

Message PWSPGSR1 indicates that the performance group service unit rate has exceeded the warning threshold;

where

mmm Is the SRM service component being monitored.

ttt Is STC, TSU, or JOB.

jjjjjjjj Is the address space name.

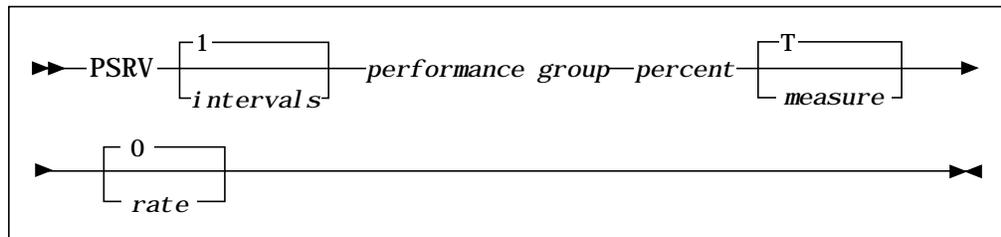
ggg Is the performance group number.

ppppp Is the number of service units.

PSRV

The Performance Group Service Rate Percentage (PSRV) sampler sends a warning message to the operator when an address space in a performance group uses more than the specified percentage of an SRM service component during a reporting interval. You can suppress the message if the total system service-rate threshold is not reached during a reporting period.

Syntax



where

<i>intervals</i>	Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).
<i>performance group</i>	Is the performance group number.
<i>percent</i>	Is the percentage of total system SRM-component use that must be reached before exception messages are issued.
<i>measure</i>	Specifies the SRM service measure to be monitored as one of the following: <ul style="list-style-type: none"> C For CPU service units I For I/O service units M For MSO service units T For total service units; the default
<i>rate</i>	Is the system SRM component use rate that must be reached before exception messages are issued. The default is a rate of 0.

Example

Assuming a report interval of 30 seconds, this control statement:

```
psrv 6, 2, 50, t, 150
```

displays warning messages at 3-minute intervals when any one address space in the performance group field uses more than 50 percent of the total SRM user rate, if the total system rate is greater than 150 SU/SEC.

Warning Messages

The following warning messages are issued by the PSRV sampler:

```
PWSPSRV0 WARNING* PERFORMANCE GROUP ggg mmm SERVICE RATE IS sssss SU/SEC
```

Message PWSPSRV0 indicates that the performance group service-unit rate has exceeded the warning threshold;

where

<i>ggg</i>	Is the performance group number.
<i>mmm</i>	Is the SRM service component being monitored.
<i>sssss</i>	Is the number of service units.

```
PWSPSRV1 WARNING* mmm SERVICE RATE FOR ttt jjjj IS pp% OF PG ggg mmm SERVICE USE
```

Message PWSPSRV1 indicates that the specified performance group service unit has exceeded the warning threshold;

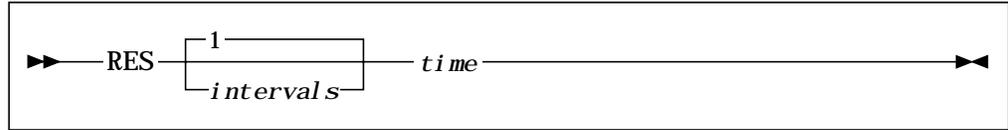
where

<i>mmm</i>	Is the SRM service component being monitored.
<i>ttt</i>	Is STC, TSU, or JOB.
<i>jjjj</i>	Is the address space name.
<i>ggg</i>	Is the performance group number.
<i>pp</i>	Is the percentage of the performance group <i>ggg</i> 's use of service component <i>mmm</i> by task <i>ttt</i>

RES

The DASD Reserves (RES) sampler monitors the reserved status of devices in a shared DASD system and notifies the operator when devices are reserved for more than a specified interval.

Syntax



where

intervals Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).

time Is the number of minutes that a device must be reserved for a warning message to be issued.

Example

Assuming a report interval of 30 seconds, this control statement:

```
res 10, 10
```

displays warning messages at 5-minute intervals if any device has been reserved for over 10 minutes.

Warning Messages

The following warning messages are issued by the RES sampler:

```
PWSRES01 WARNING* DEVICE dvn (vvvvvv) RESERVED FOR yyy MIN
```

Message PWSRES01 indicates that the device on *dvn* with volume serial number *vvvvvv* has been reserved for a period of time that exceeds the warning threshold.

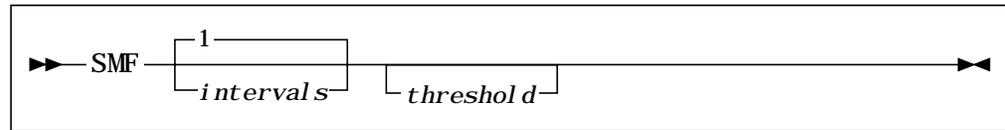
```
PWSRES01 WARNING* DEVICE cuu (vvvvvv) RESERVED FOR yyy MIN
```

Message PWSRES01 indicates that the device on *cuu* with volume serial number *vvvvvv* has been reserved for a period of time that exceeds the warning threshold.

SMF

The Status of SMF Recording (SMF) sampler notifies the operator if an SMF data set is full or SMF recording is not active, which ensures that no SMF data is lost.

Syntax



where

interval s Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).

threshold Is the percentage of space in all SMF data sets that has been filled.

Example

Assuming a report interval of 30 seconds, this control statement:

```
smf 10
```

displays a warning message every 5 minutes if any SMF data set is full.

Warning Messages

The following warning messages are issued by the SMF sampler:

```
PWSSMF01 WARNING* SMF IS NOT RECORDING
```

Message PWSSMF01 indicates that SMF data is not being collected. Accounting information will be lost.

```
PWSSMF02 WARNING* SYS1.MAN n IS FULL; DUMP IT ASAP
```

Message PWSSMF02 indicates that the indicated SMF data set is full. The SMF data set should be dumped as soon as possible so that SMF can reuse it.

```
PWSSMF03 WARNING* BOTH SMF DATA SETS ARE FULL; xxxx RECORDS LOST
```

Message PWSSMF03 indicates that SMF is unable to write records because both data sets are full; *xxxx* represents the number of records that SMF could not write.

PWSSMF04 WARNING* SYS1.MAN *n* IS PARTIALLY FULL AND INACTIVE; DUMP IT ASAP.

Message PWSSMF04 indicates that the SMF data set is partially full and inactive, and should be dumped as soon as possible so SMF can reuse it.

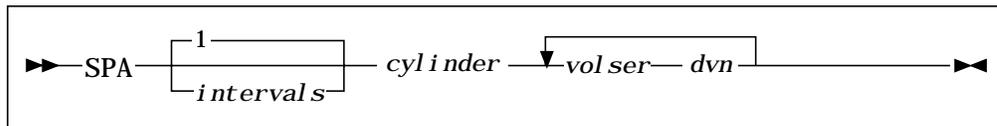
PWSSMF09 WARNING* SMF DATA SETS ARE *xxx*% FULL

Message PWSSMF09 indicates that the SMF data sets have used over a threshold percentage amount of their allocated space. You should monitor the SMF data set usage and dump them when they are filled.

SPA

The DASD Free Space (SPA) sampler monitors free space on specified DASD volumes and notifies the operator if the free space on any specified DASD volume falls below a specified number of cylinders.

Syntax



where

- interval* Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).
- cylinder* Is the low threshold of free cylinders.
- vol ser* Is the volume serial number; the maximum number of volsers you can specify on a single statement is 7. The volsers that you specify must exist at the time you invoke the sampler.
- dvn* Is the number of the device to be monitored; the maximum number of devices you can specify on a single statement is 14.

Example

Assuming a report interval of 30 seconds, this control statement:

```
spa 10, 50, work01, work02, work03, 380, 381
```

displays warning messages at 5-minute intervals if free space on WORK01, WORK02, or WORK03, or on any volume mounted on device address 380 or 381 falls below 50 cylinders.

Warning Message

The following warning message is issued by the SPA sampler:

```
PWSSPA01 WARNING* dvn vvvvvv FREE CYLINDER/TRACK COUNT IS xxx/yyyy
```

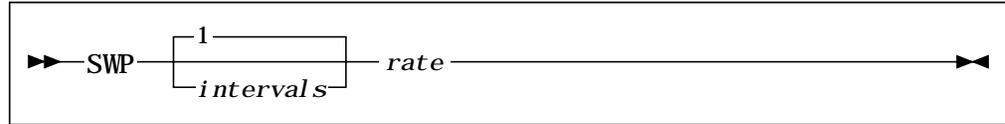
where

- dvn* Is the device address.
- vvvvvv* Is the volume serial of device.
- xxx* Is the count of free cylinders.
- yyyy* Is the count of free tracks (not including cylinders).

SWP

The Swap Rate (SWP) sampler monitors system swap-out rate and notifies the operator when the number of address spaces swapped out per minute exceeds a specified value.

Syntax



where

intervals Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).

rate Is the swap rate per minute measured over 1 reporting interval that triggers a warning message.

Example

Assuming a report interval of 30 seconds, this control statement:

```
swp 4, 30
```

displays warning messages at 2-minute intervals if the swap-out rate exceeds 30 swaps-per-minute over the preceding 2 minutes.

Warning Message

The following warning message is issued by the SWP sampler:

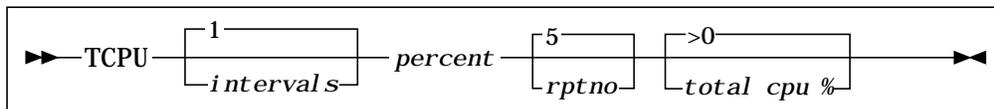
```
PWSSWP01 WARNING* SWAP OUT RATE IS xxx PER MINUTE
```

Message PWSSWP01 indicates that the swap rate has exceeded the warning threshold.

TCPU

The TSO CPU (TCPU) sampler generates a warning message whenever a TSO address space is detected using more than a specified percentage of the total CPU time during a reporting interval. The message can be suppressed if a given threshold for total CPU usage is not reached during a reporting interval. This sampler also notifies the operator when TSO address spaces are using an excessive percentage of CPU time.

Syntax



where

- intervals* Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).
- percent* Is the percentage of total system CPU time that an address space must be responsible for before an exception message is issued for that address space.
- rptno* Is the maximum number of address spaces to be listed whenever a CPU overload condition is detected. The default is 5.
- total cpu %* Is the system CPU-busy percentage that must be reached before exception messages are issued. The default is a system CPU-busy percent greater than zero.

Example

Assuming a report interval of 30 seconds, this control statement:

```
tcpu 4, 35, 5, 50
```

displays up to 5 warning messages at 2-minute intervals for each interval in which the total time the CPU is busy increases 50 percent over the preceding interval. A warning is issued for any TSO address space that has been active 35 percent of the total CPU busy time.

Warning Message

The following warning message is issued by the TCPU sampler:

```
PWSTCPU1 WARNING* CPU USAGE IS ppp% FOR TSUjjjjjjjj
```

Message PWSTCPU1 indicates that the specified address space has exceeded the warning threshold for percentage of system CPU busy time

where

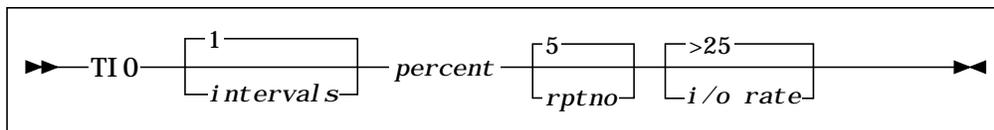
ppp Is the address space busy percentage.

jjjjjjj Is the address space name.

TIO

The TSO I/O (TIO) sampler generates a warning message whenever a specified TSO address space is detected using more than a specified percentage of the total system I/O activity in a reporting interval. The message can be suppressed if a threshold for the total system I/O rate is not reached during a reporting interval. This sampler also notifies the operator when the I/O for a particular TSO address space exceeds the threshold.

Syntax



where

<i>interval</i>	Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).
<i>percent</i>	Is the percentage of total system I/O for which a TSO address space must be responsible before an exception message is issued for that address space.
<i>rptno</i>	Is the maximum number of TSO address spaces to be listed whenever a TSO I/O overload condition is detected. The default is 5.
<i>i/o rate</i>	Is the rate of EXCPs per second that must be reached before exception messages are issued. The default is a system I/O rate greater than 25 EXCPs per second.

Example

Assuming a report interval of 30 seconds, this control statement:

```
tio 4, 25, 5, 50
```

displays up to 5 warning messages at 2-minute intervals if the system I/O rate exceeds 50 EXCPs-per-second over the preceding 2 minutes. A warning is issued for any TSO address space doing more than 25 percent of the I/O.

Warning Messages

The following warning messages are issued by the TIO sampler:

```
PWSTIO00 WARNING* TOTAL SYSTEM I/O RATE IS xxx EXCPs/SEC
```

Message PWSTIO00 indicates that the system I/O rate has exceeded the warning threshold.

```
PWSTIO01 WARNING* I/O RATE FOR TSU nnnnnnnn IS xxx% OF SYSTEM TOTAL
```

Message PWSTIO01 indicates that the specified address space has exceeded the warning threshold for percentage of system I/O;

where

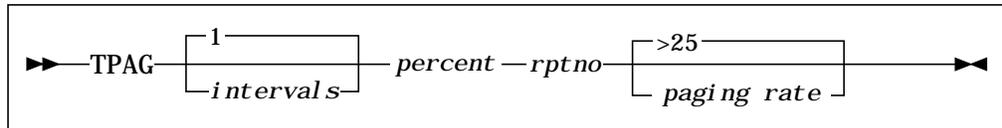
nnnnnnnn Is the address space name.

xxx Is the percent of the total system I/O rate.

TPAG

The TSO Paging (TPAG) sampler sends a warning message to the operator when a specified TSO address space is detected using more than a specified percentage of the total system paging activity during a reporting interval. The message can be suppressed if a given threshold for the total system paging rate is not reached during a reporting interval. This sampler also notifies the operator when the paging for a particular TSO address space exceeds the threshold.

Syntax



where

<i>intervals</i>	Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).
<i>percent</i>	Is the percentage of total system paging for which a TSO address space must be responsible before an exception message is issued for that address space.
<i>rptno</i>	Is the maximum number of TSO address spaces to be listed whenever a TSO-paging overload condition is detected.
<i>paging rate</i>	Is the paging rate per second that must be reached before exception messages are issued. The default is a system paging rate greater than 25 EXCPs per second.

Example

Assuming a report interval of 30 seconds, this control statement:

```
tpag 4, 25, 5, 50
```

displays up to 5 warning messages at 2-minute intervals if the system-paging rate exceeds 50 pages-per-second over the preceding 2 minutes. A warning is issued for any TSO address space doing more than 25 percent of the paging.

Warning Messages

The following warning messages are issued by the TPAG sampler:

```
PWSTPAGO WARNING* TOTAL SYSTEM PAGING RATE IS NOW ppp PAGES/SEC
```

Message PWSTPAG0 indicates that the system-paging rate has exceeded the warning threshold.

```
PWSTPAG1 WARNING* PAGING RATE FOR JOB nnnnnnnn IS xxx% OF TOTAL SYSTEM PAGING
```

Message PWSTPAG1 indicates that the specified address space has exceeded the warning threshold for percentage of system paging;

where

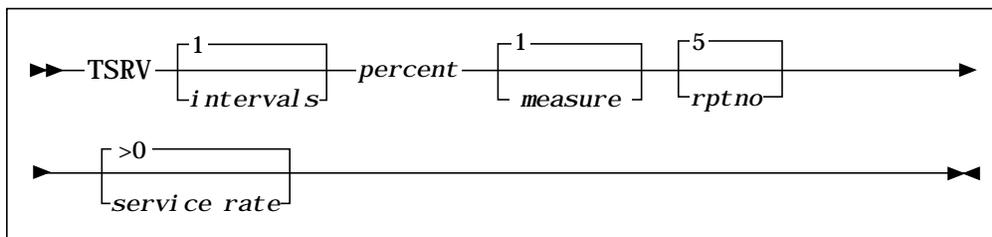
nnnnnnnn Is the address space name.

xxx Is the percent of the total system paging rate.

TSRV

The TSO Service (TSRV) sampler sends a warning message to the operator when a TSO address space uses more than the specified percentage of one or more SRM service measures during a reporting interval. The message can be suppressed if a given total system paging rate threshold is not reached during a reporting interval.

Syntax



where

<i>intervals</i>	Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).
<i>percent</i>	Is the percentage of total system SRM-component usage that TSO must be responsible for before an exception message is issued.
<i>measure</i>	Specifies the SRM service measure to be monitored as one of the following: <ul style="list-style-type: none"> C For CPU service units I For I/O service units M For MSO service units T For total service units; the default
<i>rptno</i>	Is the maximum number of address spaces to be listed whenever a TSO-service overload condition is detected. The default is 5.
<i>service rate</i>	Is the system SRM service-consumption rate that must be reached before exception messages are issued. The default is a system service rate greater than zero SUs per second.

Example

Assuming a report interval of 30 seconds, this control statement:

```
tsrv 4, 35, t, 5, 50
```

displays up to 5 warning messages at 2-minute intervals whenever an address space being monitored utilizes more than 35 percent of the total SRM components service during a reporting interval. This message can be suppressed if the system service rate threshold of 50 SUs per second is not reached during a reporting interval.

Warning Messages

The following warning messages are issued by the TSRV sampler:

```
PWSTSRV0 WARNING* ADDR SPACE mmm SERVICE RATE IS sssss SU/SEC
```

Message PWSTSRV0 indicates that the system paging rate has exceeded the warning threshold;

where

mmm Is the SRM service component being monitored.

sssss Is the number of service units per second.

```
PWSTSRV1 WARNING* mmm SERVICE RATE FOR TSUjjjjjjjj IS ppp% OF SYSTEM TOTAL
```

Message PWSTSRV1 indicates that the specified address space has exceeded the warning threshold for percentage of system SRM component utilization;

where

mmm Is the SRM service component being monitored.

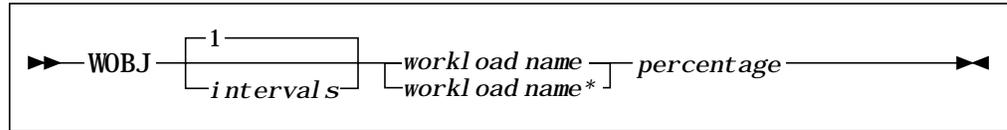
jjjjjjjj Is the address space name.

ppp Is the address space SRM component use as a percentage of system total.

WOBJ

The Workload Objective (WOBJ) sampler warns you when a workload is not meeting its specified percentage of service objectives.

Syntax



where

<i>intervals</i>	Is the number of intervals between report periods; the default is 30 seconds (1 interval).
<i>workload name</i>	Is the name of the workload to be monitored.
<i>workload name*</i>	Indicates a partial workload name. Partial names may be specified using a wildcard character (an asterisk).
<i>percentage</i>	Represents the threshold for workload service objectives, in decimal numbers. If the workload's percentage of satisfied service objectives is equal to or less than the number specified, a warning message is issued. The workload's service objectives are calculated for the current data collection interval as defined by MainView.

Example

Assuming a report interval of 30 seconds, this control statement:

```
wobj 4, mac*, 30
```

displays warning messages every 2 minutes if any workload starting with the letters MAC meets 30 percent or less of its service objectives.

Warning Message

The following warning message is issued by the WOBJ sampler:

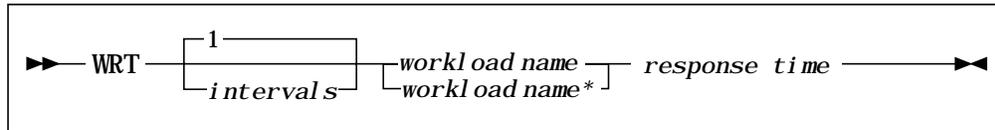
```
PWSWOBJO *WARNING* Workload wwwwww at xxx% of its objective
```

Message PWSWOBJO indicates that workload *wwwwwww* has met *xxx* percent of its objective.

WRT

The Workload Response Time (WRT) sampler warns you when a workload's response time reaches or exceeds a specified threshold.

Syntax



where

<i>intervals</i>	Is the number of intervals between report periods; the default is 30 seconds (1 interval).
<i>workload name</i>	Is the name of the workload to be monitored.
<i>workload name*</i>	Indicates a partial workload name. Partial names may be specified using a wildcard character (an asterisk).
<i>response time</i>	Represents the minimum response time, in seconds, for the workload. If the workload's response time reaches or exceeds the number of seconds specified, a warning message is issued. The response time is calculated for the current data collection interval as defined by MainView.

Example

Assuming a report interval of 30 seconds, this control statement:

```
wrt 6, mac*, 30
```

displays warning messages every 3 minutes if any workload starting with the letters MAC receives a response time of 30 seconds or greater.

Warning Message

The following warning message is issued by the WRT sampler:

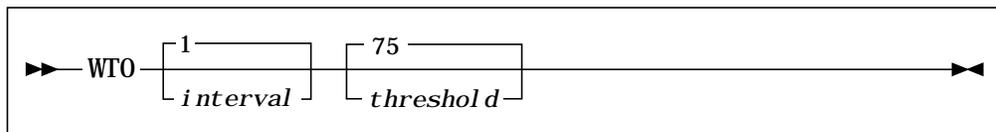
```
PWSWRT00 *WARNING* Workload wwwwww response time is xxx. xx seconds
```

Message PWSWRT00 indicates that the overall response time for workload *wwwwwww* is *xxx. xx* seconds.

WTO

The WTO Buffer Usage (WTO) sampler monitors usage of console buffers and anticipates problems caused by a backlog of messages to be displayed on a system console.

Syntax



where

interval s Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).

threshold Is the percentage of console buffers in use that causes a warning message to be issued.

Note: Default threshold is 75 percent.

Example

Assuming a report interval of 30 seconds, this control statement:

```
wto 1, 80
```

displays a warning message every 30 seconds when the use of console buffers exceeds 80 percent.

Warning Message

The following warning message is issued by the WTO sampler:

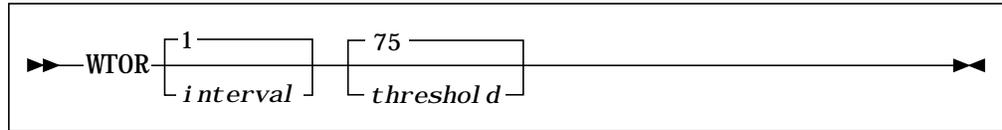
```
PWSWTO01 WARNING* CONSOLE BUFFER USAGE IS xxx%
```

Message PWSWTO01 indicates that the specified threshold has been exceeded. You should check the status of the consoles.

WTOR

The WTOR sampler monitors usage of console reply buffers and displays a warning message when the user-specified threshold is reached.

Syntax



where

interval s Specifies the length of the sample period in intervals. The default sample period is 30 seconds (1 interval).

threshold Is the percentage of console buffers in use that causes a warning message to be issued.

Note: Default threshold is 75 percent.

Example

Assuming a report interval of 30 seconds, this control statement:

```
wtor 1, 80
```

displays a warning message every 30 seconds when the use of console buffers exceeds 80 percent.

Warning Message

The following warning message is issued by the WTOR sampler:

```
PWSWTOR1 *WARNING* WTOR BUFFER USAGE IS XXX%
```

Message PWSWTOR1 indicates that the specified threshold has been exceeded. You should check the status of the consoles.

Part 4. Glossary and Index

The reference material contained in this part will help you understand the terms used in this book and locate where in the manual those terms are discussed.

Glossary	335
Index	345

Glossary

This glossary defines BMC Software terminology. Other dictionaries and glossaries may be used in conjunction with this glossary.

Since this glossary pertains to BMC Software-related products, some of the terms defined may not appear in this book.

To help you find the information you need, this glossary uses the following cross-references:

Contrast with	Indicates a term that has a contrary or contradictory meaning.
See	Indicates an entry that is a synonym or contains expanded information.
See also	Indicates an entry that contains related information.

A

action. Defined operation, such as modifying a MainView window, that is performed in response to a command. See *object*.

active window. Any MainView window in which data can be refreshed. See *alternate window*, *current window*, *window*.

administrative view. Display from which a product's management tasks are performed, such as the DSLIST view for managing historical data sets. See *view*.

ALT WIN field. Input field that allows you to specify the window identifier for an alternate window where the results of a hyperlink are displayed. See *alternate window*.

Alternate Access. See *MainView Alternate Access*.

alternate form. View requested through the FORM command that changes the format of a previously displayed view to show related information. See also *form*, *query*.

alternate window. (1) Window that is specifically selected to display the results of a hyperlink. (2) Window whose identifier is defined to the ALT WIN field. Contrast with *current window*. See *active window*, *window*, *ALT WIN field*.

analyzer. (1) Online display that presents a snapshot of status and activity data and indicates problem areas. (2) Component of CMF MONITOR. See *CMF MONITOR Analyzer*.

application. (1) Program that performs a specific set of tasks within a MainView product. (2) In MainView VistaPoint, combination of workloads to enable display of their transaction performance data in a single view.

application trace. See *trace*.

ASCH workload. Workload comprising Advanced Program-to-Program Communication (APPC) address spaces.

AutoCustomization. Online facility for customizing the installation of products. AutoCustomization provides an ISPF

panel interface that both presents customization steps in sequence and provides current status information about the progress of the installation.

automatic screen update. Usage mode wherein the currently displayed screen is refreshed automatically with new data at an interval you specify. Invoked by the ASU command.

AutoOPERATOR. MainView product that uses tools, techniques, and facilities to automate routine operator tasks and provide online performance monitoring, and that achieves high availability through error minimization, improved productivity, and problem prediction and prevention.

B

batch workload. Workload consisting of address spaces running batch jobs.

BBI. Basic architecture that distributes work between workstations and multiple MVS targets for BMC Software MainView products.

BBI-SS PAS. See *BBI subsystem product address space*.

BBI subsystem product address space (BBI-SS PAS).

MVS subsystem address space that manages communication between local and remote systems and that contains one or more of the following products:

- AutoOPERATOR
- MainView for CICS
- MainView for DB2
- MainView for DBCTL
- MainView for IMS
- Command MQ for S/390
- MainView VistaPoint (for CICS, DB2, and IMS workloads)

BBPARM. See *parameter library*.

BBPROC. See *procedure library*.

BBPROF. See *profile library*.

BBSAMP. See *sample library*.

BBV. See *MainView Alternate Access*.

BBXS. BMC Software Subsystem Services. Common set of service routines loaded into common storage and used by several BMC Software MainView products.

border. Visual indication of the boundaries of a window.

bottleneck analysis. Process of determining which resources have insufficient capacity to provide acceptable service levels and that therefore can cause performance problems.

C

CAS. Coordinating address space. One of the address spaces used by the MainView windows environment architecture. The CAS supplies common services and enables communication between linked systems. Each MVS image requires a separate CAS. Cross-system communication is established through the CAS using VTAM and XCF communication links.

CFMON. See *coupling facility monitoring*.

chart. Display format for graphical data. See also *graph*.

CICSplex. User-defined set of one or more CICS systems that are controlled and managed as a single functional entity.

CMF MONITOR. Comprehensive Management Facility MONITOR. Product that measures and reports on all critical system resources, such as CPU, channel, and device usage; memory, paging, and swapping activity; and workload performance.

CMF MONITOR Analyzer. Batch component of CMF MONITOR that reads the SMF user and 70 series records created by the CMF MONITOR Extractor and/or the RMF Extractor and formats them into printed system performance reports.

CMF MONITOR Extractor. Component of CMF that collects performance statistics for CMF MONITOR Analyzer, CMF MONITOR Online, MainView for MVS, and RMF postprocessor. See *CMF MONITOR Analyzer*, *CMF MONITOR Online*, *MainView for MVS*.

CMF MONITOR Online. Component of CMF that uses the MainView window interface to present data on all address spaces, their use of various system resources, and the delays that each address space incurs while waiting for access to these resources. See *CMF MONITOR*, *MainView for MVS*.

CMF Type 79 API. Application programming interface, provided by CMF, that provides access to MainView SMF-type 79 records.

CMFMON. Component of CMF MONITOR that simplifies online retrieval of information about system hardware and

application performance and creates MainView SMF-type 79 records.

The CMFMON *online facility* can be used to view data in one or more formatted screens.

The CMFMON *write facility* can be used to write collected data as MainView SMF-type 79 records to an SMF or sequential data set.

CMRDETL. MainView for CICS data set that stores detail transaction records (type 6E) and abend records (type 6D). Detail records are logged for each successful transaction. Abend records are written when an abend occurs. Both records have the same format when stored on CMRDETL.

CMRSTAT. MainView for CICS data set that stores both CICS operational statistic records, at 5-minute intervals, and other records, at intervals defined by parameters specified during customization (using CMRSOFT).

column. Vertical component of a view or display, typically containing fields of the same type of information, that varies by the objects associated in each row.

collection interval. Length of time data is collected. See also *delta mode*, *total mode*.

command delimiter. Special character, usually a ; (semicolon), used to stack commands typed concurrently on the COMMAND line for sequential execution.

COMMAND line. Line in the control area of the display screen where primary commands can be typed. Contrast with *line command column*.

Command MQ Automation D/S. Command MQ agents, which provide local proactive monitoring for both MQSeries and MSMQ (Microsoft message queue manager). The Command MQ agents operate at the local node level where they continue to perform functions regardless of the availability of the MQM (message queue manager) network. Functionality includes automatic monitoring and restarts of channels, queue managers, queues and command servers. In cases where automated recovery is not possible, the agents transport critical alert information to a central console.

Command MQ Automation S/390. Command MQ component, which monitors the MQM (message queue manager) networks and intercedes to perform corrective actions when problems arise. Solutions include:

- Dead-Letter Queue management
- System Queue Archival
- Service Interval Performance solutions
- Channel Availability

These solutions help ensure immediate relief to some of the most pressing MQM operations and performance problems.

Command MQ for D/S. Command MQ for D/S utilizes a true client/server architecture and employs resident agents to provide configuration, administration, performance monitoring and operations management for the MQM (message queue manager) network.

Command MQ for S/390. Delivers comprehensive capabilities for configuration, administration, performance monitoring and operations management for an entire MQM (message queue manager) network.

COMMON STORAGE MONITOR. Component of MainView for MVS that monitors usage and reconfigures MVS common storage blocks.

composite workload. Workload made up of a WLM workload or other workloads, which are called *constituent workloads*.

constituent workload. Member of a composite workload. Constituent workloads in a composite usually belong to a single workload class, but sometimes are mixed.

contention. Occurs when there are more requests for service than there are servers available.

context. In a Plex Manager view, field that contains the name of a target or group of targets specified with the CONTEXT command. See *scope*, *service point*, *SSI context*, *target context*.

CONTEXT command. Specifies either a MainView product and a specific target for that product (see *target context*) or a MainView product and a name representing one or more targets (see *SSI context*) for that product.

control statement. (1) Statement that interrupts a sequence of instructions and transfers control to another part of the program. (2) Statement that names samplers and other parameters that configure the MainView components to perform specified functions. (3) In CMF MONITOR, statement in a parameter library member used to identify a sampler in the extractor or a report in the analyzer, or to describe either component's processing requirements to the operating system.

coupling facility monitoring (CFMON). Coupling facility views that monitor the activity of your system's coupling facilities.

CPO. Customized Product Offering. Delivery and installation technique that allows any combination of BMC Software SMP/E-maintainable products to be distributed on a product tape to a customer and installed quickly. The CPO product tape contains libraries required for product customization and execution, plus SMP distribution libraries and data sets needed for application of SMP maintenance.

current data. Data that reflects the system in its current state. The two types of current data are realtime data and interval data. Contrast with *historical data*. See also *interval data* and *realtime data*.

current window. In the MainView window environment, window where the main dialog with the application takes place. The current window is used as the default window destination for commands issued on the COMMAND line when no window number is specified. Contrast with *alternate window*. See *active window*, *window*.

D

DASD. Direct Access Storage Device. (1) A device with rotating recording surfaces that provides immediate access to stored data. (2) Any device that responds to a DASD program.

DASD ADVISOR. An interactive software tool that diagnoses DASD performance problems and makes recommendations to reduce overall service time. This tool measures and reports on the operational performance of IBM and IBM-compatible devices.

data collector. Program that belongs to a MainView product and that collects data from various sources and stores the data in records used by views. For example, MainView for MVS data collectors obtain data from MVS services, MVS control blocks, CMF MONITOR Extractor control blocks, and other sources. Contrast with *extractor*.

delta mode. (1) In MainView for DB2 analyzer displays, difference between the value sampled at the start of the current statistics interval and the value sampled by the current analyzer request. See also *statistics interval*. (2) In CMFMON, usage mode wherein certain columns of data reflect the difference in values between one sample cycle and the next. Invoked by the DELTA ON command. See also *collection interval*, *sample cycle*, *total mode*.

DMR. See *MainView for DB2*.

DSO. Data Set Optimizer. CMF MONITOR Extractor component that uses CMF MONITOR Extractor data to produce reports specifying the optimal ordering of data sets on moveable head devices.

E

element. (1) Data component of a data collector record, shown in a view as a field. (2) Internal value of a field in a view, used in product functions.

element help. Online help for a field in a view. The preferred term is *field help*.

Event Collector. Component for MainView for IMS and MainView for DBCTL that collects data about events in the IMS environment. This data is required for Workload Monitor and optional for Workload Analyzer (except for the workload trace service). This data also is recorded as transaction records (X'FA') and program records (X'F9') on the IMS system log for later use by the batch products: Performance Reporter and Transaction Accountant. If the Extensions for DB2 option to IMS PR, IMS TA, IMS WA, or IMS WM is installed, the Event Collector also measures DB2 activity through the Attach facility.

expand. Predefined link from one display to a related display. See also *hyperlink*.

Extensions for DB2. Additions to MainView for IMS and MainView for DBCTL that gather DB2 subsystem activity through the IMS Attach facility. These additions are licensed as options to IMS PR, IMS TA, IMS WA, and IMS WM. No

license is required for IMS RA, IMS RM, and AutoOPERATOR for IMS.

Extensions for IRLM. Additions to MainView for IMS and MainView for DBCTL that analyze database locking and measure IRLM activity when IRLM is used. These additions are licensed as options to IMS RA and IMS RM.

extractor. Program that collects data from various sources and keeps the data control blocks to be written as records. Extractors obtain data from services, control blocks, and other sources. Contrast with *data collector*.

extractor interval. See *collection interval*.

F

fast path. Predefined link between one screen and another. To use the fast path, place the cursor on a single value in a field and press Enter. The resulting screen displays more detailed information about the selected value. See also *hyperlink*.

field. Group of character positions within a screen or report used to type or display specific information.

field help. Online help describing the purpose or contents of a field on a screen. To display field help, place the cursor anywhere in a field and press PF1 (HELP). In some products, field help is accessible from the screen help that is displayed when you press PF1.

filter. Selection criteria used to limit the number of rows displayed in a view. Data that does not meet the selection criteria is not displayed. A filter is composed of an element, an operator, and an operand (a number or character string). Filters can be implemented in view customization, through the PARM/QPARM commands, or through the Where/QWhere commands. Filters are established against elements of data.

fixed field. Field that remains stationary at the left margin of a screen that is scrolled either right or left.

FOCAL POINT. MainView product that displays a summary of key performance indicators across systems, sites, and applications from a single terminal.

form. One of two constituent parts of a view; the other is *query*. A form defines how the data is presented; a query identifies the data required for the view. See also *query*, *view*.

full-screen mode. Display of a MainView product application or service on the entire screen. There is no window information line. Contrast with *windows mode*.

G

global command. Any MainView window interface command that can affect all windows in the window area of a MainView display.

graph. Graphical display of data that you select from a MainView window environment view. See also *chart*.

H

hilevel. For MainView products, high-level data set qualifier required by a site's naming conventions.

historical data. (1) Data that reflects the system as it existed at the end of a past recording interval or the duration of several intervals. (2) Any data stored in the historical database and retrieved using the TIME command. Contrast with *current data*, *interval data* and *realtime data*.

historical database. Collection of performance data written at the end of each installation-defined recording interval and containing up to 100 VSAM clusters. Data is extracted from the historical database with the TIME command. See *historical data*.

historical data set. In MainView products that display historical data, VSAM cluster file in which data is recorded at regular intervals.

hyperlink. (1) Preset field in a view or an EXPAND line on a display that permits you to

- Access cursor-sensitive help
- Issue commands
- Link to another view or display

The transfer can be either within a single product or to a related display/view in a different MainView product. Generally, hyperlinked fields are highlighted. (2) Cursor-activated short path from a topic or term in online help to related information. See also *fast path*.

I

Image log. Collection of screen-display records. Image logs may be created for both the BBI-SS PAS and the BBI terminal session (TS).

The BBI-SS PAS Image log consists of two data sets that are used alternately: as one fills up, the other is used. Logging to the BBI-SS PAS Image log stops when both data sets are filled and the first data set is not processed by the archive program.

The TS Image log is a single data set that wraps around when full.

IMS PERFORMANCE REPORTER (IMS PR). Offline product that organizes data and prints reports that can be used to analyze IMS performance.

IMS PR EXTENSIONS for DB2. Licensed option to the IMS Performance Reporter that provides offline statistical, graphic, and calendar reports about DB2 subsystem activity and requests that are integrated with IMS activity.

IMS RA EXTENSIONS for DB2. Additions to the IMS Resource Analyzer that provide online monitoring displays of DB2 region connection and thread status.

IMS RESOURCE ANALYZER (IMS RA). Online realtime displays used to analyze IMS resources and determine which are affected by specific workload problems.

IMS RESOURCE MONITOR (IMS RM). Online data collection services used to monitor IMS resources and issue warnings when defined utilization thresholds are exceeded.

IMS RM EXTENSIONS for DB2. Additions to the IMS Resource Monitor that provide timer-driven data collection of DB2 region connection and threads.

IMS TA EXTENSIONS for DB2. Licensed option to the IMS Transaction Accountant that provides integrated accounting of IMS and DB2 activity through the IMS Attach Facility.

IMS TRANSACTION ACCOUNTANT (IMS TA). Offline product used to produce cost accounting and user charge-back records and reports.

IMS WA EXTENSIONS for DB2. Licensed option to the IMS Workload Analyzer that provides

- Timer-driven workload wait and trace data collection
- Displays of the transaction active and wait time for DB2 processing events
- Trace of DB2 subsystem activity

The trace is either summarized by DB2 calls and CPU usage for DB2 processing or detailed to include call start, elapsed times, SQL statement numbers, and return codes.

IMS WORKLOAD ANALYZER (IMS WA). Online data collection and display services used to analyze IMS workloads and determine problem causes.

IMS WM EXTENSIONS for DB2. Licensed option to the IMS Workload Monitor that provides timer-driven data collection of the types and number of calls issued to a DB2 subsystem, DB2 transaction input queue and response time, and average DB2 CPU time per transaction.

IMS WORKLOAD MONITOR (IMS WM). Online data collection services used to monitor IMS workloads and issue warnings when defined thresholds are exceeded.

IMSPlex System Manager (IPSM). Online service that provides Single System Image views of resources and bottlenecks for applications across one or more IMS regions and systems.

interval data. Cumulative data collected during a collection interval. Intervals usually last from 15 to 30 minutes depending on how the recording interval is specified during product customization. Contrast with *historical data*.

Note: If change is made to the workloads, a new interval will be started.

See also *current data and realtime data*.

InTune. Product for improving application program performance. It monitors the program and provides information used to reduce bottlenecks and delays.

IRUF. IMS Resource Utilization File (IRUF). IRUFs can be either detailed (one event, one record) or summarized (more than one event, one record). A detailed IRUF is created by processing the IMS system log through a program called IMFLEDIT. A summarized IRUF is created by processing one

or more detailed IRUFs, one or more summarized IRUFs, or a combination of both, through a sort program and the TASCOSTR program. If the Extensions for DB2 option to IMS PR or IMS TA is installed, the IRUF includes data about DB2 calls made through the IMS Attach Facility.

J

job activity view. Report about address space consumption of resources. See *view*.

journal. Special-purpose data set that stores the chronological records of operator and system actions.

Journal log. Collection of messages. Journal logs are created for both the BBI-SS PAS and the BBI terminal session (TS).

The BBI-SS PAS Journal log consists of two data sets that are used alternately: as one fills up, the other is used. Logging to the BBI-SS PAS Journal log stops when both data sets are filled and the first data set is not being processed by the archive program.

The TS Journal log is a single data set that wraps around when full.

L

line command. Command that you type in the line command column in a view or display. Line commands initiate actions that apply to the data displayed in that particular row.

line command column. Command input column on the left side of a view or display. Contrast with *COMMAND line*.

log edit. In the MainView for IMS and MainView for DBCTL program named IMFLEDIT, function that extracts transaction (X'FA') and program (X'F9') records from the IMS system log. IMFLEDIT also extracts certain records that were recorded on the system log by IMS. IMFLEDIT then formats the records into a file called the IMS Resource Utilization File (IRUF).

M

MainView. BMC Software integrated systems management architecture.

MainView Alarm Manager. Monitor that reads the data elements produced by products in the MainView window environment and returns SQL-syntactic statements.

MainView Alternate Access. Enables MainView products to be used without TSO by providing access through EXCP and VTAM interfaces.

MainView control area. In the MainView window environment, first three lines at the top of the view containing the window information line and the COMMAND, SCROLL, CURR WIN, and ALT WIN lines. The control area cannot be customized and is part of the information display. Contrast with *MainView display area, MainView window area*.

MainView Desktop. Version of the MainView window interface designed to run on OS/2 and Windows workstations.

MainView display area. See *MainView window area*.

MainView Explorer. Product that provides access to MainView products from a Web browser running under Windows. MainView Explorer replaces MainView Desktop.

MainView for CICS. Product (formerly MV MANAGER for CICS) that provides realtime application performance analysis and monitoring for CICS system management.

MainView for DB2. Product (formerly MV MANAGER for DB2) that provides realtime and historical application performance analysis and monitoring for DB2 subsystem management.

MainView for DBCTL. Product (formerly MV MANAGER for DBCTL) that provides realtime application performance analysis and monitoring for DBCTL management.

MainView for IMS. Product (formerly MV MANAGER for IMS) that provides realtime application performance analysis and monitoring for IMS management.

MainView for MVS. System management application (formerly MV MANAGER for MVS). Built upon the MainView window environment architecture, it uses the window interface to provide access to system performance data and other functions necessary in the overall management of an enterprise.

MainView for OS/390. System management application (formerly MainView for MVS (prior to version 2.5)). Built upon the MainView window environment architecture, it uses the window interface to provide access to system performance data and other functions necessary in the overall management of an enterprise.

MainView for USS. System management application that allows you to monitor the performance of the Unix System Services from a MainView window interface.

MainView Selection Menu. ISPF selection panel that provides access to all MainView windows-mode and full-screen mode products.

MainView VistaPoint. Product that provides enterprise-wide views of performance. Application and workload views are available for CICS, DB2, IMS, and MVS. Data is summarized at the level of detail needed; e.g., reports may be for a single target, an MVS image, or an entire enterprise.

MainView window area. Portion of the information display that is not the control area and in which views are displayed and windows opened. It includes all but the first three lines of the information display. Contrast with *MainView control area*.

monitor. Online service that measures resources or workloads at user-defined intervals and issues warnings when user-defined thresholds are exceeded.

MV MANAGER for CICS. See *MainView for CICS*.

MV MANAGER for DB2. See *MainView for DB2*.

MV MANAGER for DBCTL. See *MainView for DBCTL*.

MV MANAGER for IMS. See *MainView for IMS*.

MV MANAGER for MVS. See *MainView for MVS*.

MVALARM. See *MainView Alarm Manager*.

MVCICS. See *MainView for CICS*.

MVDB2. See *MainView for DB2*.

MVDBC. See *MainView for DBCTL*.

MVIMS. See *MainView for IMS*.

MVMQS. See *Command MQ for S/390*

MVMVS. See *MainView for OS/390*.

MVS product address space (PAS). Address space containing MVS data collectors, including the CMF MONITOR Extractor. Used by MainView for MVS and CMF MONITOR products. See *PAS*.

MVScope. MainView for MVS application that traces both CPU usage down to the CSECT level and I/O usage down to the channel program level.

MVVP. See *MainView VistaPoint*.

N

nested help. Multiple layers of help pop-up windows. Each successive layer is accessed by hyperlinking from the previous layer.

O

object. Anything you can manipulate as a single unit. MainView objects can be any of the following: product, secondary window, view, row, column, or field.

You can issue an action against an object by issuing a line command in the line command column to the left of the object. See *action*.

OMVS workload. Workload consisting of MVS OpenEdition address spaces.

online help. Help information that is accessible online.

P

parameter library. Data set comprised of members containing parameters for specific MainView products or a support component. There can be several versions:

- The distributed parameter library, called BBPARM
- A site-specific parameter library or libraries

These can be

- A library created by AutoCustomization, called UBBPARAM
- A library created manually, with a unique name

PAS. Product address space. Used by the MainView products. Contains data collectors and other product functions. See *MVS product address space (PAS)*, *BBI subsystem product address space (BBI-SS PAS)*.

performance group workload. MVS/SP-defined collection of address spaces. See *service class workload*, *workload definition*.

PERFORMANCE MANAGER. MainView for CICS online service for monitoring and managing current performance of CICS regions.

PERFORMANCE REPORTER. Product component that generates offline batch reports. The following products can generate these reports:

- MainView for DB2
- MainView for IMS
- MainView for CICS

Plex Manager. Product through which cross-system communication, MainView security, and an SSI context are established and controlled. Plex Manager is shipped with MainView window environment products as part of the coordinating address space (CAS) and is accessible as a menu option from the MainView Selection Menu.

pop-up window. Window containing help information that, when active, overlays part of the window area. A pop-up panel is displayed when you issue the HELP command.

PRGP workload. In MVS/SP 5.0 or earlier, or in compatibility mode in MVS/SP 5.1 or later, composite of service classes. MainView for MVS creates a performance group workload for each performance group defined in the current IEAIPsxx member.

procedure library. Data set comprised of members containing executable procedures used by AutoOPERATOR. These procedures are execute command lists (EXECs) that automate site functions. There can be several versions:

- The distributed parameter library, called BBPROC
- A site-specific parameter library or libraries

These can be

- A library created by AutoCustomization, called UBBPROC
- A library created manually, with a unique name

The site-created EXECs can be either user-written or customized AutoOPERATOR-supplied EXECs from BBPROC.

product address space. See *PAS*.

profile library. Data set comprised of members containing profile information and cycle refresh definitions for a terminal session connected to a BBI-SS PAS. Other members are

dynamically created by MainView applications. There can be several versions:

- The distributed profile library, called BBPROF
- A site-specific profile library or libraries

These can be

- A library created by AutoCustomization, called SBBPROF
- A library created manually, with a unique name

The site library is a common profile shared by all site users. The terminal session CLIST creates a user profile automatically if one does not exist; it is called `userid.BBPROF`, where `userid` is your logon ID. User profile libraries allow each user to specify unique PF keys, CYCLE commands, target system defaults, a Primary Option Menu, and a unique set of application profiles.

Q

query. One of two constituent parts of a view; the other is *form*. A query defines the data for a view; a form defines the display format. See also *form*, *view*.

R

realtime data. Performance data as it exists at the moment of inquiry. Realtime data is recorded during the smallest unit of time for data collection. Contrast with *historical data*. See also *current data and interval data*.

row. (1) Horizontal component of a view or display comprising all the fields pertaining to a single device, address space, user, etc. (2) Horizontal component of a DB2 table consisting of a sequence of values, one for each column of the table.

RxD2. Product that provides access to DB2 from REXX. It provides tools to query the DB2 catalog, issue dynamic SQL, test DB2 applications, analyze EXPLAIN data, generate DDL or DB2 utility JCL, edit DB2 table spaces, perform security administration, and much more.

S

sample cycle. Time between data samples.

For the CMF MONITOR Extractor, this is the time specified in the extractor control statements (usually 1 to 5 seconds).

For realtime data, the cycle is not fixed. Data is sampled each time you press Enter.

sample library. Data set comprised of members each of which contains one of the following:

- Sample JCL that can be edited to perform specific functions
- A macro that is referenced in the assembly of user-written services

- A sample user exit routine

There can be several versions:

- The distributed sample library, called BBSAMP
- A site-specific sample library or libraries

These can be

- A library created by AutoCustomization, called UBBSAMP
- A library created manually, with a unique name

sampler. Program that monitors a specific aspect of system performance. Includes utilization thresholds used by the Exception Monitor. The CMF MONITOR Extractor contains samplers.

SBBPROF. See *profile library*.

scope. Subset of an SSI context. The scope could be all the data for the context or a subset of data within the context. It is user- or site-defined. See *SSI context, target*.

screen definition. Configuration of one or more views that have been stored with the SAVEScr command and assigned a unique name. A screen includes the layout of the windows and the view, context, system, and product active in each window.

selection view. In MainView products, view displaying a list of available views.

service class workload. MVS- or MainView for MVS-defined collection of address spaces.

If you are running MVS Workload Manager (WLM) in goal mode, MainView for MVS creates a service class workload for each service class that you define through WLM definition dialogs.

If you are running MVS 4.3 or earlier, or MVS/SP 5.1 or later with WLM in compatibility mode, MVS creates a performance group workload instead of a service class. See *performance group workload*.

service objective. Workload performance goal, specified in terms of response time for TSO workloads or turnaround time for batch workloads. Performance group workloads can be measured by either objective. Composite workload service objectives consist of user-defined weighting factors assigned to each constituent workload. There are no MVS-related measures of service for started task workloads.

service point. Specification, to MainView, of the services required to enable a specific product. Services may be actions, selectors, or views. Each target (e.g., CICS, DB2, or IMS,) has its own service point.

The PLEX view lists all the defined service points known to the CAS to which the terminal session is connected.

service request block (SRB). Control block that represents a routine to be dispatched. SRB mode routines generally perform work for the operating system at a high priority. An SRB is similar to a task control block (TCB) in that it identifies a unit of work to the system. See also *task control block*.

service select code. Code entered to invoke analyzers, monitors, and general services. This code is also the name of the individual service.

session. Total period of time an address space has been active. A session begins when monitoring can be performed. If the product address space (PAS) starts after the job, the session starts with the PAS.

SRB. See *service request block*.

single system image (SSI). Feature of the MainView window environment architecture that allows you to view and perform actions on multiple MVS systems as though they were a single system. The rows of a single tabular view can contain rows from different MVS images.

SpaceView for MVS. Suite of products that assist in all phases of MVS storage management. SpaceView consists of components that perform automation, reporting, trend analysis, and error correction for storage management in MVS.

SSI. See *single system image*.

SSI context. Name created to represent one or more targets for a given product. See *context, target*.

started task workload. Address spaces running jobs that were initiated programmatically.

statistics interval. For MainView for DB2, cumulative count within a predefined interval (30-minute default set by the DB2STATS parameter in the distributed BBPARM member BBIISP00) for an analyzer service DELTA or RATE display. Specifying the DELTA parameter displays the current value as the difference between the value sampled by the current analyzer request and the value sampled at the start of the current interval. Specifying the RATE parameter displays the current value by minute (DELTA divided by the number of elapsed minutes).

summary view. View created from a tabular view using the Summarize option in view customization. A summary view compresses several rows of data into a single row based on the summarize criteria.

system resource. See *object*.

T

target. Entity monitored by one or more MainView products, such as an MVS image, IMS or DB2 subsystem, CICS region, or related workloads across systems. See *context, scope, SSI context*.

target context. Single target/product combination. See *context*.

TASCOSTR. MainView for IMS and MainView for DBCTL program that summarizes detail and summary IMS Resource Utilization Files (IRUFs) to be used as input to the offline products.

task control block (TCB). Address space-specific control block that represents a unit of work that is dispatched in the address space in which it was created. See also *service request block*.

TCB. See *task control block*.

terminal session (TS). Single point of control for MainView products, allowing data manipulation and data display and providing other terminal user services for MainView products. The terminal session runs in a user address space (either a TSO address space or a standalone address space for EXCP/VTAM access).

TDIR. See *trace log directory*.

threshold. Specified value used to determine whether the data in a field meets specific criteria.

TLDS. See *trace log data set*.

total mode. Usage mode in CMFMON wherein certain columns of data reflect the cumulative value between collection intervals. Invoked by the DELTA OFF command. See also *collection interval, delta mode*.

trace. (1) Record of a series of events chronologically listed as they occur. (2) Online data collection and display services that track transaction activity through DB2, IMS, or CICS.

trace log data set (TLDS). Single or multiple external VSAM data sets containing summary or detail trace data for later viewing or printing. The trace log(s) can be defined as needed or dynamically allocated by the BBI-SS PAS. Each trace request is assigned its own trace log data set(s).

trace log directory (TDIR). VSAM linear data set containing one entry for each trace log data set. Each entry indicates the date and time of data set creation, the current status of the data set, the trace target, and other related information.

transaction. Specific set of input data that initiates a predefined process or job.

TS. See *terminal session*.

TSO workload. Workload that consists of address spaces running TSO sessions.

U

UAS. See *user address space*.

UBBPARM. See *parameter library*.

UBBPROC. See *procedure library*.

UBBSAMP. See *sample library*.

user address space. Runs a MainView terminal session (TS) in TSO, VTAM, or EXCP mode.

User BBPROF. See *profile library*.

V

view. Formatted data within a MainView window, acquired from a product as a result of a view command or action. A view consists of two parts: query and form. See also *form, job activity view, query*.

view definition. Meaning of data that appears online, including source of data, selection criteria for data field inclusion and placement, data format, summarization, context, product, view name, hyperlink fields, and threshold conditions.

view command. Name of a view that you type on the COMMAND line to display that view.

view command stack. Internal stack of up to 10 queries. For each command, the stack contains the filter parameters, sort order, context, product, and timeframe that accompany the view.

view help. Online help describing the purpose of a view. To display view help, place the cursor on the view name on the window information line and press PF1 (HELP).

W

window. Area of the MainView screen in which views and resources are presented. A window has visible boundaries and can be smaller than or equal in size to the MainView window area. See *active window, alternate window, current window, MainView window area*.

window information line. Top border of a window. Shows the window identifier, the name of the view displayed in the window, the system, the scope, the product reflected by the window, and the timeframe for which the data in the window is relevant. See also *window status field*.

window number. Sequential number assigned by MainView to each window when it is opened. The window number is the second character in the window status field. See also *window status field*.

window status. One-character letter in the window status field that indicates when a window is ready to receive commands, is busy processing commands, is not to be updated, or contains no data. It also indicates when an error has occurred in a window. The window status is the first character in the window status field. See also *window information line, window status field*.

window status field. Field on the window information line that shows the current status and assigned number of the window. See also *window number, window status*.

windows mode. Display of one or more MainView product views on a screen that can be divided into a maximum of 20 windows. A window information line defines the top border of each window. Contrast with *full-screen mode*.

WLM workload. In goal mode in MVS/SP 5.1 and later, a composite of service classes. MainView for MVS creates a

workload for each WLM workload defined in the active service policy.

workflow. Measure of system activity that indicates how efficiently system resources are serving the jobs in a workload.

workload. (1) Systematic grouping of units of work (e.g., address spaces, CICS transactions, IMS transactions) according to classification criteria established by a system administrator. (2) In MVS, group of service classes within a service definition.

workload activity view. Tracks workload activity as the workload accesses system resources. A workload activity view measures workload activity in terms of resource consumption and how well the workload activity meets its service objectives.

workload definition. Workload created through the WKLIST view. Contains a unique name, a description, an initial status, a current status, and selection criteria by which address spaces are selected for inclusion in the workload. See *Workload Definition Facility*.

Workload Definition Facility. In MainView for MVS, WKLIST view and its associated dialogs through which workloads are defined and service objectives set.

workload delay view. Tracks workload performance as the workload accesses system resources. A workload delay view measures any delay a workload experiences as it contends for those resources.

workload objectives. Performance goals for a workload, defined in WKLIST. Objectives may include measures of performance such as response times and batch turnaround times.

Index

Symbols

\$\$INEXCL 7, 8
 \$\$INLGxx 127, 299
 \$\$INSYS0 7
 parameters 10
 \$\$INTKxx 235
 \$\$INWTO 7, 9
 \$\$PRJOBS 7, 15
 \$\$RPTEXT 7, 16
 \$\$XENQ 7, 17

A

accessing SYSPROG services
 from an operator console 31
 from easy menus 23
 from EZMFPROG 24
 from EZMSPROG 23
 from MAIN view 25
 from SYSPROG Services Menu panel 25
 with a CMDID character 31
 with the MODIFY command 31
 ACPU sampler 254
 action 282
 active task queue
 monitoring 279
 address space
 abnormally terminated 224
 displaying
 allocated devices 226
 device usage 245
 frame allocation 184
 LSQA usage 170, 175
 RBs 221
 status 214
 step information 132, 175
 storage allocations by subpool and key 171
 TCBs 221
 TQEs 231
 trace buffer information 237
 listing regions 174
 making nonswappable 73
 making swappable 160
 monitoring
 activity 150
 vector table slots 305
 queues maintained in ASVT 50
 summarizing virtual storage allocations 169
 SVC dump 188
 terminating 103
 address spaces
 monitoring 264, 266
 CPU usage 254
 EXCPs issued 258
 I/O usage 256

 number in use 320
 paging activity 260, 262
 real storage usage 316
 AIO sampler 256
 AIOR sampler 258
 ALLOC service 40
 APAG sampler 260
 APF list
 adding entries 44
 changing the volser of existing entries 44
 changing to DYNAMIC format 45
 deleting entries 44
 displaying the format 46
 listing entries 43
 restoring the list created at IPL 45
 updating 41
 APF service 41
 APGR sampler 262
 ASM service 47
 ASRV sampler 264
 ASSR sampler 266
 ASVT service 50
 authorization lists
 AUTHCMD 51
 AUTHPGM 51
 AUTHTSF 51
 building 53
 displaying 52
 restoring 53
 AUTHTSO service 51

B

batch address spaces
 monitoring
 CPU usage 268
 I/O usage 269
 paging activity 271
 SRM component usage 273
 BBILIB member
 SYSPROGJ 32
 BBPARM library members
 \$\$INSYS0 117
 LLA keyword 117
 \$\$INTKxx 235
 executing commands listed in 101
 PWSCPMxx
 setting report and sample intervals 293
 BBXS service
 displaying status 55
 BCPU sampler 268
 BIO sampler 269
 BPAG sampler 271
 BSRV sampler 273

C

- CDE service 57
- CHA sampler 275
- channel paths
 - monitoring 144, 150, 153, 275
- CHAP service 59
- CLEAR service 61
- CMDID characters
 - using 31
- COMMAND service 63
- command syntax 29
- commands
 - Execute 27
 - Locate 26
 - Select 26
- common service area
 - See CSA storage
- common storage
 - monitoring usage
 - by jobname 282
- component usage
 - monitoring performance groups 310
- console buffers
 - monitoring 333, 334
- CONSOLES service 64
- Contents Directory Entry (CDE)
 - listing for address spaces 57
- control blocks
 - ASVT 50
 - LPA active queue 126
 - SRM 209
 - Subsystem Vector Table (SSVT) 212
- CP commands
 - entering 248
- CPU busy percentage
 - displaying by domain 177
 - displaying by partition 177
- CPU model number 106
- CPU queue depth
 - monitoring 279
- CPU sampler 277
- CPU serial number 106
- CPU service 65
- CPU usage
 - displaying by jobname 65
 - monitoring address spaces 254
 - monitoring batch address spaces 268
 - monitoring by system 277
 - monitoring TSO address spaces 322
 - non-MDF systems 66
 - non-PR/SM systems 66
- CQD sampler 279
- CSA sampler 280
- CSA service 68
- CSA storage
 - displaying usage 68
 - monitoring usage 280
 - by jobname 282
- CSECT

- displaying 158
- CSMJ sampler 282
- CSVLLAyy 114
- CSVLLAzz 114

D

- DASD volumes 80, 287
 - displaying
 - available space 207
 - data blocks by cylinder, track, and record number 81
 - data blocks by data set name 82
 - data set names 83
 - extent list 83
 - Format 4 DSCB 82
 - label data block 84
 - pending mount requests 157
 - reserve activity 182
 - VTOC entry 83
 - evaluating contention 147
 - measuring head movement 146
 - monitoring
 - available space 319
 - I/O queue depth 285
 - pending mount requests 302
 - reserve activity 315
 - mounting 152
 - not ready 287
 - under control of SMS 201
- data sets
 - displaying information about 74
 - usage by address space 226
- date
 - displaying 112, 225
- DCPU service 177
- DD names
 - listing members within a concatenation 104, 226
- DEALLOC service 70
- DEV sampler 284
- DEVIATN service 71
- devices
 - allocated to an address space 226
 - allocating 40
 - deallocating 70
 - identifying excessive contention 284
 - monitoring 150, 284
 - submitting 217
 - use by address space 245
- DIOQ sampler 285
- Disabled Interrupt Exit (DIE) routines
 - execution 231
 - pending 231
- dispatching priorities
 - changing for an address space 59
- DNR sampler 287
- DOM sampler
 - monitoring MPL 288
- domain
 - monitoring 288

- domains
 - displaying processor usage 177
- DONTSWAP service 73
- DSNAME service 74
- DUMP service 76
- DVIEW service 80
- DVTN sampler 289

E

- easy menus
 - EZM390 23
 - EZMFPROG 24
 - EZMSPROG 23
- ECSA storage
 - displaying usage 68
 - monitoring usage
 - by jobname 282
- EDTINFO service 85
- elapsed time
 - monitoring jobs 294
- Eligible Device Table (EDT) 85
- ENQ sampler 291
- enqueue conflicts
 - monitoring 291
- ENQUEUES
 - displaying conflicts 89
 - excluding names from display 17
 - service 89
- entering commands 29
- entry point addresses
 - displaying 158
- EQUATE service 92
- ESCLASS service 94
- esoteric unitnames 85
- ESQA storage
 - monitoring usage by jobname 282
- ESTORAGE service 99
- Exception Monitor
 - setting the report interval 293
 - setting the sample interval 293
- Exception Monitor services
 - controlling 249
 - excluding job names from 8
 - storing as members in BBPARM 18
- EXCPs
 - monitoring address spaces 258
- EXECUTE service 101
- executing commands 101
- exempting from reply scan 15
- exit routines 197
- EXIT service 103
- expanded storage
 - displaying information about 99
- extended common service area
 - See ECSA storage
- extended storage
 - See expanded storage
- Extended SVC Router (ESR) table 126

F

- FINDMBR service 104
- FIX sampler 292
- fixed link-pack area (FLPA) 130
- fixed pages
 - monitoring 292
- frame allocation
 - displaying for an address space 184
 - monitoring usage by job 296

G

- generic unitnames 85
- glossary 337
- Greenwich Mean Time (GMT)
 - displaying 225

I

- I/O interrupt activity
 - displaying distribution among processors 164
- I/O queue depth
 - monitoring DASD devices 285
 - monitoring page data sets 309
- I/O subsystem
 - monitoring 144
- I/O usage
 - monitoring address spaces 256
 - monitoring batch address spaces 269
 - monitoring TSO address spaces 324
- IEFPARM 114
- IKJTABLS
 - corresponding entry points in IKJTSoxx 54
- INFO service 106
- INT service 293
- IO service 107
- IOS driver codes 108
- IOS drivers 229
- IPLDATA service 109
- ISPF EDIT 89

J

- JET sampler 294
- jobs
 - awaiting V=R region 298
 - frame usage 296
 - monitoring elapsed time 294
 - slot usage 296
 - swapped out of memory 307
- JSU sampler 296
- JVR sampler 298

L

- LABEL service 111
- LCPU service 112
- link list
 - displaying information about 114

- for OS/390 1.3 118
- management 114
- rebuilding dynamically 114
- link-pack areas
 - locating modules in 130
- list of services 25
- LLA parameter 114
- LLIST service 114
- LNKLST service 118
 - OS/390 1.3 support 118
- LOADLPA service 124
- LOG service 127, 299
- logical control units
 - monitoring 147
- logical partitions (LPARs)
 - displaying processor usage 177
- LOGREC data sets
 - reading software records 203
- LPA active queue 126
- LPA service 130
- LWAIT service 132
- LX service 134

M

- main memory
 - modifying 251
- major name 89
- master console
 - issuing commands from a TSO terminal 136
- MCOMMAND service 136
- MDEVICE service 137
- memory locations
 - displaying 76
 - searching for strings 139
- MEMSCAN service 139
- minor name 89
- MIO service 144
- MLCU service 147
- modified link-pack area (MLPA) 130
- modules
 - adding in LPA 124
 - deleting 124
 - displaying LOGREC data 203
 - listing for address spaces 57
 - locating in system link-pack areas 130
- MONITOR service 150
- monitoring
 - domain MPL 288
 - enqueue conflicts 291
- mount requests 157, 302
- MOUNT service 152
- MP environments 67
- MPATH service 153
- MSTORAGE service 155
- MTP sampler 302
- MTP service 157
- Multiple Domain Facility (MDF) complex 177
- multiprogramming level (MPL) 288

N

- non-MDF systems 66
- non-PR/SM systems 66
- non-TP I/O
 - displaying 107
- non-VIO slots 47
- NRQP sampler 305
- NUCLEUS service 158

O

- OKSWAP service 160
- operating system
 - displaying information about 106
- OUT sampler 307
- outstanding replies
 - displaying 181, 314
 - monitoring 314

P

- PAG sampler 308
- page fixing
 - monitoring 292
- page reclaim rate 162
- paging activity
 - analyzing 162
 - monitoring address spaces 260, 262
 - monitoring batch address spaces 271
 - monitoring rate 308
 - monitoring TSO address spaces 326
- paging data sets 47
 - monitoring I/O queue depth 309
- PAGING service 162
- parameters
 - define run-time 10
- PARMLIST service 163
- PCPU service 177
- pending mount requests
 - displaying 157
 - monitoring 302
- performance group 310
 - SRM component usage 312
- PGQ sampler 309
- PGSR sampler 310
- PIO service 164
- PPT service 165
- PR/SM systems
 - displaying CPU usage information 65
- PRIVATE service 169
- program properties table
 - updates 165
- PROGRESS service 175
- PRSM service 177
- PSRV sampler 312
- PWSCPM 7
- PWSCPMxx 18

Q

Q name

See major name

queues

ASVT control block 50

R

R name

See minor name

RBs

displaying internal values 221

real storage

displaying allocation 184

monitoring status 316

Recovery Termination Manager (RTM) 103

relative share 66

REP sampler 314

replies

monitoring 314

REPLIES service 181

reply buffers

listing number in use 64

request blocks

See TCBs

RES sampler 315

reserve activity

displaying for DASD devices 182

monitoring DASD devices 315

RESERVES service 182

resource ownership information

displaying 91

RSM service 184

RSU sampler 316

S

scanning memory 139

SDUMP service 188

SETOP service 191

slots

monitoring usage by job 296

SMF

data sets 317

monitoring status 317

recording 317

sampler 317

SMF exits 192, 197

deleting 193, 197, 198

disabling 195

displaying status 194

enabling 193, 195, 198

loading 195

routines 197

SMFEXIT service 192, 197

SMSLIST service 201

SOFTFRR service 203

software LOGRECs

summarizing 203

SPA sampler 319

SPACE service 207

SQA storage

displaying usage 68

monitoring usage 280

by jobname 282

SRB mode 79

SRM 310

component usage 264, 266

monitoring address spaces 264, 266

monitoring batch address spaces 273

monitoring TSO address spaces 328

sampler 320

SRM service 209

SSVT service 212

STATUS service 214

SUBMIT service 217

Subsystem Communications Vector Table (SSCVT)

mapping information from 212

suppressing WTOR message 9

SVC dump

clearing 61

displaying date and title 220

taking 188

SVC table 126

scanning 218

SVCFIND service 218

swap-out rate 307

monitoring 321

SWP sampler 321

symbols

defining 92, 139

displaying 92, 139

syntax diagrams 29

reading viii

SYS1.LINKLIB 116

SYS1.PARMLIB members

CONFIGxx 71, 289

CSVLLAy 114

IEAAPFxx 43

IEASYS00 320

IKJTSoxx 51

listing contents 163

SYSDUMP service 220

SYSEVENT 316

SYSPROG

independent operations 32

started task (STC) 32

TSO command 32

SYSPROG services 112

accessing from easy menus 21

ALLOC 40

APF 41

ASM 47

ASVT 50

AUTHTSO 51

BBXS 55

CDE 57

CHAP 59

- CLEAR 61
 - COMMAND 63
 - CONSOLES 64
 - CPU 65
 - CSA 68
 - DEALLOC 70
 - DEVIATN 71
 - DONTSWAP 73
 - DSNAME 74
 - DUMP 76
 - DVIEW 80
 - EDTINFO 85
 - ENQUEUEES 89
 - entering commands 29
 - EQUATE 92
 - ESCLASS 94
 - ESTORAGE 99
 - EXECUTE 101
 - EXIT 103
 - FINDMBR 104
 - INFO 106
 - introduction 3
 - IO 107
 - IPLDATA 109
 - LABEL 111
 - LLIST 114
 - LNKLST 118
 - LOADLPA 124
 - LPA 130
 - LWAIT 132
 - LX 134
 - MCOMMAND 136
 - MDEVICE 137
 - MEMSCAN 139
 - MIO 144
 - MLCU 147
 - MONITOR 150
 - MOUNT 152
 - MPATH 153
 - MSTORAGE 155
 - MTP 157
 - NUCLEUS 158
 - OKSWAP 160
 - PAGING 162
 - PARMLIST 163
 - PIO 164
 - PPT 165
 - PRIVATE 169
 - PROGRESS 175
 - PRSM 177
 - REPLIES 181
 - RESERVES 182
 - RSM 184
 - SDUMP 188
 - SETOP 191
 - SMFEXIT 192, 197
 - SMSLIST 201
 - SOFTFRR 203
 - SPACE 207
 - SRM 209
 - SSVT 212
 - STATUS 214
 - SUBMIT 217
 - SVCFIND 218
 - SYSDUMP 220
 - TCB 221
 - TERM 224
 - TIME 225
 - TIOT 226
 - TPIO 229
 - TQE 231
 - TRACK 234
 - TRTBL 237
 - TSULIST 240
 - UCB 242
 - USING 21, 245
 - VIO 247
 - VMVMD 248
 - WARNING 249
 - ways to access 21
 - ZAP 251
 - SYSPROG Services Menu panel 25
 - system console
 - listing number of undisplayed messages 64
 - system dump data sets 188
 - clearing 61
 - displaying date and title 220
 - system linkage indexes 134
 - system problems
 - solving 63
 - system queue area
 - See* SQA storage
 - system resource information
 - displaying 209
 - system-managed storage (SMS)
 - displaying DASD information 201
- ## T
- tape labels
 - displaying contents 111
 - task control block
 - See* TCBS
 - TCB service 221
 - TCBs
 - displaying internal values 221
 - displaying storage allocations by subpool and key 173
 - TCPU sampler 322
 - teleprocessing devices
 - displaying I/O 229
 - TERM service 224
 - Terminal I/O Task service 226
 - terminating address space 103
 - time
 - displaying 225
 - TIME service 225
 - Timer Queue Elements (TQEs)

- displaying for an address space 231
- TIO sampler 324
- TIOT service 226
- TPAG sampler 326
- TPIO service 229
- TQE service 231
- trace buffers
 - summarizing 237
 - TTE type 237
- TRACK service 234
- TRTBL service 237
- TSO
 - See also* authorization lists
 - session information 240
- TSO address spaces
 - monitoring
 - CPU usage 322
 - I/O usage 324
 - paging activity 326
 - SRM
 - component usage 328
- TSRV sampler 328
- TSULIST service 240

U

- UCB service 242
- unit control blocks (UCB)
 - displaying 242
- unitnames
 - displaying information about 85
- Unreferenced Interval Count (UIC) 320
- USING service 245

V

- V=R region
 - monitoring waiting jobs 298
- VIO service 247
- VIO slots 47
 - displaying usage by jobname or user ID 247
- virtual console interface 248
- virtual storage areas
 - displaying usage 155
 - mapping 155
- VMVMD service 248
- volumes
 - DASD
 - free space (SPA) sampler 319
- VSAM data sets 75

W

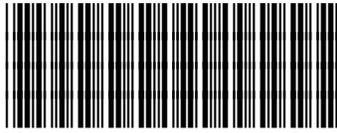
- wait state 64
- WARNING service 249
- WDLY service 310, 330
- workload
 - monitoring delays 330
 - monitoring response time 332

- monitoring service objectives 331
- WRT sampler 332
- WTO sampler 333, 334
- WTORs
 - exempting from reply scan 16

Z

- ZAP service 251

Notes



100025940