STORAGETEK®

PART NUMBER	VERSION NUMBER	REVISION
311290608	3.1	J

SVAA

SHARED VIRTUAL ARRAY ADMINISTRATOR

REPORTING FOR OS/390

PRODUCT TYPE
SOFTWARE

Shared Virtual Array Administrator

Version 3.1

for OS/390

Reporting

Part Number: 311290608

We welcome your feedback. Please contact the Global Learning Solutions Feedback System at:

GLSFS@Sun.com

or

Global Learning Solutions
Sun Microsystems Inc.
One StorageTek Drive
Louisville, CO 80028-3256
USA

Contents

Figures

Tables

About This Book

Shared Virtual Array Administrator (SVAA) helps you configure and administer the Shared Virtual Array (SVA) subsystem and produce reports about the SVA's performance. SVAA also can help you manage certain other DASD subsystems[1] by reporting on their efficiency and degree of utilization. This book describes how to use the Reporter component of SVAA.

Note: Although this book generally mentions only OS/390, the SVAA for OS/390 software runs identically on **z/OS, OS/390, and MVS host operating systems**.

Who Should Read This Book

This book is for people responsible for DASD administration, capacity planning, and performance analysis. In particular, it is for those wanting to obtain reports about the performance of SVA and certain other DASD subsystems.

This book presumes that readers are familiar with the operation of the SVA subsystem and with measuring of the performance of DASD subsystems.

Organization of This Book

This book is organized as follows:

Chapter 1, Introduction	Introduces the concept of subsystem reporting as a part of SVAA.
Chapter 2, Overview of SVAA Subsystem Reporting	Introduces the three major functions of Reporter.
Chapter 3, Reporter in the OS/390 Environment	Describes running Reporter on OS/390 systems.
Chapter 4, Controlling Data Collection	Describes how to control data collection.
Chapter 5, Managing the Collected Data	Describes how to manage the collected data.
Chapter 6, Controlling the Reports	Describes how to control the output reports.
Chapter 7, Using the SVAA Panels	Describes how to use interactive Reporter panels to control data collection and reporting.
Chapter 8, SVAA Subcommands and Commands	Describes how to use Reporter subcommands to control collection and reporting, either from the command line or from a batch program. The chapter also describes how to specify the logging files and how to manage the collected data.
Chapter 9, Understanding the SVAA Reports	Describes how to interpret the output reports.

[1] Subsystems attached to controllers that are compatible with the IBM 3990.

Chapter 10, Format of Logging Records	Shows the format of Reporter records.
Appendix A, Sample OS/390 Job Control Language	Lists the JCL needed to run Reporter under OS/390.
Appendix B, Exception Thresholds	Describes the exception thresholds and their defaults.
Appendix C, Variables for Bar Chart Reports	Describes the fields that can be requested for a bar chart report.
Appendix D, Information in Host Records	Describes the variables whose values can be requested for the different kinds of reports, and which reports the variables appear in.
Appendix E, Using Data Extraction	Provides suggestions for creating graphs of extracted data and lists the variables that can be specified in data extraction.

A glossary, a list of acronyms and abbreviations, and an index appear at the back of the book.

Conventions Used in This Book

This book uses the following conventions in describing the subcommands and panels:

- Commands and values that you enter are shown in bold type (for example, **SET ECAMDEV**).
- Uppercase letters in subcommand or parameter names indicate the required letters; the lowercase letters can be omitted. For example, **SASSubsystemFiLe** shows that **SASSFL** is the short form of the **SASSUBSYSTEMFILE** parameter.
- In the panel descriptions, field names and key names are shown in bold type (for example, the **Dates** field, the **Enter** key).

For more details of the conventions, see “Subcommand Descriptions” on page 8-10.

Notation Used in Syntax Diagrams

Throughout this library, diagrams are used to illustrate the programming syntax. The following list tells you how to interpret the syntax diagrams:

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

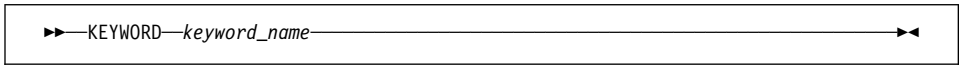
  The ▸▸— symbol indicates the beginning of a statement.

  The —▸ symbol indicates that the statement syntax continues on the next line.

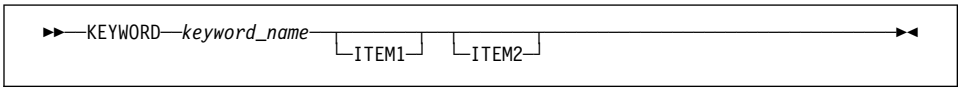
  The ▸— symbol indicates that a statement is continued from the previous line.

  The —▸◂ symbol indicates the end of a statement.

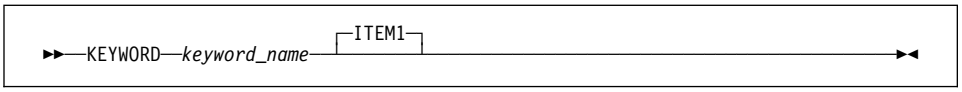
- Items shown on the main path of the statement are required.



- Items shown on branches below the main path are optional.

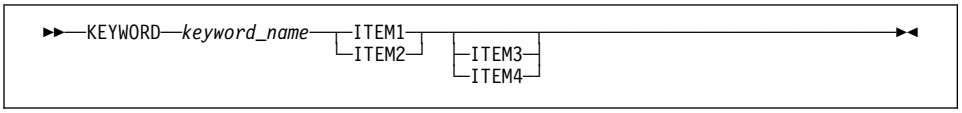


- Items shown on branches above the main path are default values.



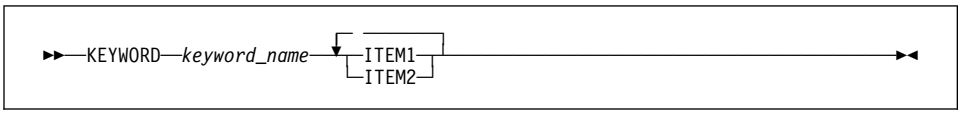
- Items appearing in a stack indicate that only one of the items can be specified. When one of the items in a stack appears on the main path, you must include one of the items.

  For example, in the following diagram, you must include either ITEM1 or ITEM2. ITEM3 and ITEM4 both appear below the main path, so neither one is required.

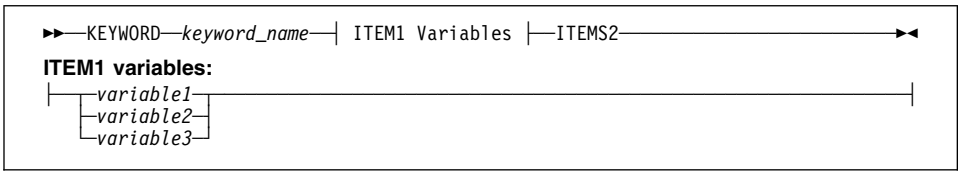


- A repeat arrow shown above an item or a stack of items indicates that you can specify the item multiple times or specify more than one of the items. A character (such as a comma or a blank) on the repeat arrow indicates that the items must be separated by that character.

  For example, in the following syntax diagram, you can specify both ITEM1 and ITEM2, but you must use a blank to separate your choices in your programming syntax.



- In some cases, when an item has additional items associated with it, an additional syntax diagram is shown that represents the full syntax of that item. For example, in the following syntax diagram, additional information that can or must be specified for ITEM1 appears in the “ITEM1 Variables” syntax diagram.



Shared Virtual Array Documentation

This section lists both software documentation and hardware documentation for the Shared Virtual Array products.

How to Obtain Software Documentation

All of the Shared Virtual Array software publications are available:

- On the "SVA Software Publications" CD-ROM (part number 3134524*nn*). To order a copy, contact StorageTek Publication Sales and Service at 800-436-5554 or send a fax to 303-661-7367.
- Online (for viewing and printing), at the StorageTek Customer Resource Center (CRC) web site at: **www.support.storagetek.com**
  Click on Software and go to the Shared Virtual Array Software list.

  Note: Access to the CRC site requires a password. To obtain a password, call StorageTek Customer Support at 800-678-4430.

SVA Administrator Library:

SVA Administrator for OS/390

- *Shared Virtual Array Administrator for OS/390 Configuration and Administration*
  3112905*nn*
- *Shared Virtual Array Administrator for OS/390 Installation, Customization, and Maintenance*
  3112908*nn*
- *Shared Virtual Array Administrator for OS/390 Reporting*
  3112906*nn*
- *Shared Virtual Array SnapShot for OS/390 Installation, Customization, and Maintenance*
  3112913*nn*

SVA Administrator for VM

- *Shared Virtual Array Administrator for VM Configuration and Administration*
  3134629*nn*
- *Shared Virtual Array Administrator for VM Installation, Customization, and Maintenance*
  3134631*nn*
- *Shared Virtual Array Administrator for VM Reporting*
  3134630*nn*

SVA Administrator for OS/390 and VM

- *Shared Virtual Array Administrator for OS/390 and VM Messages and Codes*
  3112907*nn*

For any StorageTek Software:

- *Requesting Help from Software Support*
  1121240*nn*

SVA Hardware Publications

Shared Virtual Array hardware publications are available:

- Online (for viewing and printing), at the StorageTek Customer Resource Center (CRC) web site at: **www.support.storagetek.com**
  Click on Disk Subsystems.

  Note: Access to the CRC site requires a password. To obtain a password, call StorageTek Customer Support at 800-678-4430.

V2Xf SVA Library:

- *V2Xf Shared Virtual Array General Information*
  MO9216*x*
- *V2Xf Shared Virtual Array Introduction*
  MO9217*x*
- *V2Xf Shared Virtual Array Operation and Recovery*
  MO9219*x*
- *V2Xf Shared Virtual Array Planning*
  MO9218*x*
- *V2Xf Shared Virtual Array Reference*
  MO9220*x*
- *V2Xf Shared Virtual Array System Assurance*
  MO9221*x*
- *V2Xf Shared Virtual Array Peer-to-Peer Remote Copy Configuration Guide (PPRCfcn)*
  MO9211*x*

V2X SVA Library:

- *V2X Shared Virtual Array General Information*
  MO9133*x*
- *V2X Shared Virtual Array Introduction*
  MO9135*x*
- *V2X Shared Virtual Array Operation and Recovery*
  MO9137*x*
- *V2X Shared Virtual Array Planning*
  MO9136*x*

- *V2X Shared Virtual Array Reference*
  MO9139*x*
- *V2X Shared Virtual Array System Assurance*
  MO9138*x*
- *V2X Shared Virtual Array System Assurance*
  MO9138*x*

V960 SVA Library:

- *V960 Shared Virtual Array General Information*
  MO5011*x*
- *V960 Shared Virtual Array Introduction*
  MO5006*x*
- *V960 Shared Virtual Array Operation and Recovery*
  MO5007*x*
- *V960 Shared Virtual Array Planning*
  MO5008*x*
- *V960 Shared Virtual Array Reference*
  MO5009*x*
- *V960 Shared Virtual Array System Assurance*
  MO5010*x*

Peer-to-Peer Remote Copy for V2X, V2X2, and V960:

- *Peer-to-Peer Remote Copy Configuration Guide*
  MP4007*x*

Trademarks

The following terms are trademarks or registered trademarks of Storage Technology Corporation:

- Iceberg
- Shared Virtual Array
- SnapVantage
- StorageTek
- SVA

The following terms are trademarks or registered trademarks of International Business Machines Corporation:

- DFSMS/MVS
- DFSMS/VM
- ESCON
- FICON
- IBM
- MVS
- OS/390
- RACF
- RMF
- VM/ESA
- z/OS
- z/VM

The following terms are trademarks or registered trademarks of Microsoft Corporation:

- DOS
- Excel
- Windows

The following terms are trademarks or registered trademarks of SAS Institute, Incorporated:

- SAS
- SAS/C
- SAS/GRAPH

The following terms are trademarks of Sun Microsystems, Inc.:

- Solaris
- Java

HP-UX is a trademark of Hewlett-Packard Company.

UNIX is a registered trademark of The Open Group.

Note: Other terms and product names used herein may be trademarks of their respective owners.

Product Support

StorageTek Customer Services provides 24-hour assistance for questions or problems related to StorageTek products. Calls from our customers receive immediate attention from trained diagnostic specialists. Call 800-678-4430.

Refer to the document *Requesting Help from Software Support* for detailed information about contacting StorageTek for technical support in your country or geographical location.

During problem resolution, Software Support may request that you provide specific diagnostic materials. Although printed data might be acceptable, data in machine-readable form is much preferred.

OS/390 Diagnostic Materials

Software Support may request one or more of the following kinds of diagnostic materials, depending on the nature of the problem:

- Details of circumstances
- OS/390 SYSLOG
- SYSMSG data set
- SYSPRINT data set
- SYS*x*DUMP and SYS1.DUMP*nn* data sets
- SYSHIST data set (LOG OFFLOAD)
- Database DUMPS/DEBUG
- EREP records (hardware and/or software)
- ISPF panel images
- ISPF panel names and SPFLOG
- External trace for SVAA via GTF
- CCW I/O traces
- SMF records
- Listings of SVAA files altered during installation, including the PROFSIBS and PROFSIBA macros
- Copies of logging files
- Output of SVAA started-task job
- Console dump with: SDATA=(ALLPSA,SQA,LSQA,RGN,LPA, TRT,CSA,SWA,SUMDUMP,ALLNUC,Q=YES,GRSQ)

Summary of Changes

Revision J, May 2006 -- EC 132661

This revision:

- Adds support for the Capacity on Demand (COD) feature.

All significant changes are identified by a vertical bar in the left margin.

Ninth Edition (Rev I), March 2005 -- EC 132026

This edition:

- Adds 3390-9 to the list of possible values for DEVICE TYPE in the Definition of Field Names section of Chapter 9.

All significant changes are identified by a vertical bar in the left margin.

Eighth Edition (Rev H), April 2004 -- EC 128971

This edition:

- Updates the QUERY DEVICE subcommand to indicate that the subcommand does not display information about PAV Alias devices.
- Updates the SET ECAMDEVICE subcommand to note that PAV Alias devices can not be defined as ECAM devices.
- Adds notes that PAV Alias devices are excluded from all data collection. As a result, performance and space utilization data is not reported for Aliases.

All significant changes are identified by a vertical bar in the left margin.

Seventh Edition (Rev G), November 2003 -- EC 128861

This edition:

- Uses the term “FICON” where appropriate.
- Uses the term “ESCON” rather than “serial.”
- Updates the “chanType” field of the Channel Interface
- Updates the “chanType” field of the Channel Interface Statistics Record in Chapter 10.
- Adds explanations for the functional capacity of SCSI devices to the Definitions of Field Names in Chapter 9.
- Makes minor corrections and edits throughout the document.

All significant changes are identified by a vertical bar in the left margin.

Sixth Edition (Rev F), February 2003 -- EC 128687

This edition

- Adds a new appendix, Appendix F, which describes the point-in-time reports.
- Makes minor corrections and edits throughout the document.

All significant changes are identified by a vertical bar in the left margin.

Fifth Edition (Rev E), September 2002 -- EC 128567

This edition:

- Adds a SNAPSHOT option to the SVAA main menu (page 7-5). (This enables you to navigate to the SnapShot ISPF panels, which are described in the *SVAA for OS/390 Configuration and Administration* manual.
- Revises unit and volume parameter descriptions in Chapter 8 to reflect that you can define up to 4096 devices on the V2X SVA.
- Corrects some field descriptions in the SMF Subsystem Performance Record in Chapter 10.
- Corrects some SAS label descriptions in in Table D-1 in Appendix D.
- Makes minor corrections and edits throughout the document.

All significant changes are identified by a vertical bar in the left margin.

Fourth Edition (Rev D), April 2001 -- EC 123253

This edition does not contain technical changes or enhancements to the SVAA product. It does incorporate many corrections and improvements in the text.

Third Edition (Rev C), October 2000 -- EC 123170

This edition:

- Includes notes about the implications of Power PPRC for data collection and reporting:
  - Performance data is not collected for Power PPRC Bridge devices (pages 2-8, 4-11, and in several subcommand descriptions in Chapter 8).
  - In the SAS Macro library, Power PPRC Bridge devices are considered CKD devices by data collection and reporting (pages 2-8 and 4-11).
  - The QUERY DEVICE subcommand (Chapter 8) cannot be used to display information about Power PPRC Bridge devices.
  - Power PPRC Bridge devices cannot be defined as ECAM devices (SET ECAMDEVICE subcommand in Chapter 8).
  - Power PPRC Bridge devices must not be used to store user data. Therefore, generating Space Utilization for such devices serves no purpose (REPORT SPACEUTILIZATION subcommand in Chapter 8).
- Adds a Device Type column to the SELECT DEVICES panels (Chapter 7).

- Updates some report examples in Chapter 9.
- Adds two PPRC volume types to the Functional Device Performance SMF record and the Space Utilization SMF record (Chapter 10).
- Corrects the field type for numerous fields in the Space Utilization SMF record (Chapter 10).
- Incorporates many minor corrections and clarifications.

Chapter 1. Introduction

Shared Virtual Array Administrator (SVAA) is host software that helps you manage the Shared Virtual Array (SVA) subsystem. SVAA runs on several operating systems—including OS/390. SVAA provides facilities for administering, configuring, and obtaining reports about the performance of SVA subsystems, about the performance of traditional DASD subsystems, and about the cache effectiveness of certain non-SVA subsystems.[1]

Reporter is the part of SVAA that collects data from your subsystems and produces reports based upon that data. The Reporter program enables you to understand how well the SVA subsystems are performing, how well they are utilizing space, and how effectively they are using the cache.

This manual describes the reporting functions of SVAA Reporter. For documentation of the administration and configuration functions of SVAA see *SVAA for OS/390 Configuration and Administration*.

With Reporter, you can control which statistics from the subsystems are reported, the time intervals that the reports cover, the times for which reports are to be produced, and the contents of the reports that Reporter produces. Alternatively, you can use data collected from the subsystems as input to your own reporting programs.

Reporter consists of three functions:

- Collecting subsystem data
- Managing the collected data
- Reporting on the data

Chapter 2 introduces the three functions and the control you have over them; later chapters describe the three processes in detail.

Figure 1-1 on page 1-2 shows a simplified view of the three functions of Reporter, with the input and output of each function.

[1] Reporter also provides Cache Effectiveness reports for subsystems attached to controllers that are compatible with the IBM 3990.

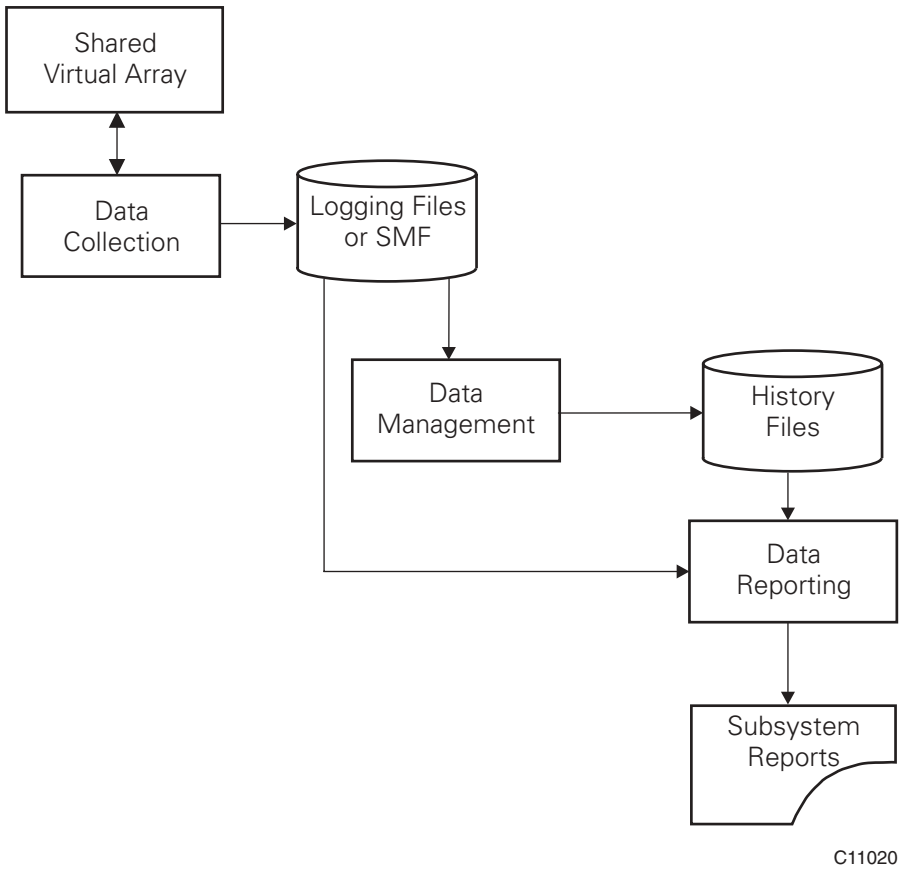


Figure 1-1. *Simplified view of Reporter*

You can access the Reporter functions by:

- Using SVAA panels under ISPF
- Entering SVAA commands on the TSO command line
- Entering SVAA commands from a CLIST or REXX EXEC.
- Submitting SVAA commands as a batch job
- Entering SVAA operator commands from the host console

Chapter 2. Overview of SVAA Subsystem Reporting

Chapter Summary

This chapter provides an overview of the three major functions of Reporter:

1. Collecting the data
2. Managing the collected data
3. Reporting on the data

Collecting Subsystem Data

This section describes the data you can collect from the subsystems, and explains what you can control about collection so that the reports based on the collected data will be most useful.

Chapter 4, “Controlling Data Collection” provides details of specifying data collection.

Categories of Data

Reporter can collect both performance data and space-utilization data.

Performance Data

An SVA subsystem gathers statistics about its operation at intervals. Reporter commands initiate collection of statistics from the subsystems and determine how the statistics are stored for later analysis.

You can collect data for analysis of:

- Performance of the functional devices
- Cache effectiveness
- Space utilization (net capacity load and free-space collection load)

Reporter collects this data at the times you specify and stores the data in DASD files.

Space-Utilization Data

Unlike data about performance and space utilization, data about space utilization is a view of the conditions at a particular time. While performance data is collected at intervals and reported on later, space-utilization data is used directly to produce reports.

Data about utilization of space includes:

- The amount of functional space defined
- The amount of functional space allocated
- The amount of functional space used
- The amount of physical back-end space used by each functional volume

Controlling Data Collection

This section summarizes the aspects of collection over which you have control with Reporter. You control the times for collection and the devices for which statistics are to be collected. Each time Reporter collects statistics from a subsystem is called a *collection instance*.

Collecting Performance-Tracking Data

Typically, statistics collected with Reporter are *performance-tracking* (PT) data.

Performance-tracking (PT) data is a set of statistics about subsystem performance gathered over a period of time, at specified intervals. You can specify the collection of either device-performance data or cache-effectiveness data, or both. The Data Collection Task collects PT data at intervals and accumulates the data in the main logging file. There can be only one PT data collection run in a Data Collection Task at any one time.

Once you start data collection, Reporter collects data indefinitely (or until you stop collection). PT data consists of statistics collected for a defined period, as follows:

- Device performance, including:
  - Access density
  - Device utilization
  - I/O response time
  - I/O rates (by channel, subsystem, and functional device)
  - Channel activity
  - Free space collection load
  - Net capacity load
  - Drive module utilization (physical devices)
- Cache effectiveness, including:
  - I/O rates (for the subsystems and devices)
  - Read, write, and I/O cache hit percentages
  - DASD fast-write nonvolatile storage constraint
  - Stages per second
  - Hits per stage
  - Low reference count
  - Track occupancy

Reporter either stores the collected PT data in a sequential file called the *main logging file* or sends the data to the System Management Facility (SMF), whichever you specify. If you are not using SMF and if you anticipate collecting large amounts of data from your DASD, you can define an additional file, an *alternate logging file*, so that when one file becomes full, collection can be switched to the other, as shown in Figure 2-1.

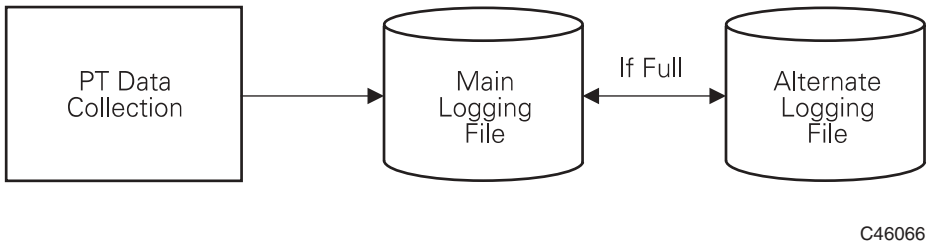


Figure 2-1. *Full PT logging file*

The logging files are stored on DASD. The data is stored in the logging files in the same format as that of SMF records.

Collecting On-Request Data

At times, you may want to make an ad hoc study of the performance of a DASD subsystem or change the parameters you specified for PT data collection. Rather than changing the parameters of PT data collection, you can request a special collection run that focuses more closely on the devices or time periods in which you are interested. Such collection is called *on-request* (OR) data collection. (Unlike PT data collection, you can have multiple OR data collection runs at the same time.)

OR data collection is used to gather the same kinds of data as performance-tracking data collection, but with different collection parameters. For example, OR collection can be used to gather data about a specific aspect of subsystem performance, in a more specific way than in PT collection. Again, you may notice in a cache-effectiveness exception report that the values of a variable have changed significantly. To collect statistics for that subsystem over more-frequent intervals, you can use OR data collection to gather the needed information without disturbing the normal collection of PT data for the subsystem. As further examples, you can request a special data collection run to help in diagnosing DASD performance problems, or to obtain detailed analysis with finer time resolution than you specified for PT data collection.

As with PT data collection, you can specify one or two files for data collection, a *primary OR output file* and an optional *alternate OR output file*. When one file becomes full, collection is switched to the other, as shown in Figure 2-2. OR data collection has no effect on PT data collection.

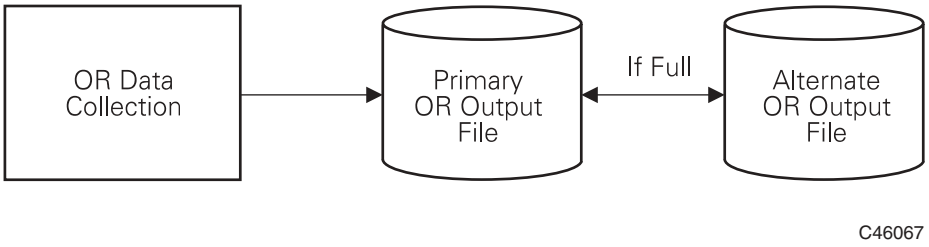


Figure 2-2. *Full OR logging file*

You can collect the OR data in the same files as the PT data by giving the OR collection run a unique identifier to distinguish the resulting data from the PT data.

Controlling When to Collect Data

You control, for each day, the time ranges over which collection is active, how often collection is made, and the synchronization of collection.

Controlling Collection Ranges

With Reporter you have great flexibility in specifying *ranges of times* (or *time ranges*) when data is to be collected. You can specify up to 16 time ranges for each day's data collection. These ranges are broad periods (usually hours) during which data is to be collected. For example, you can specify that an overall range of time for data collection is from 8 a.m. to 9 a.m. and from 10 a.m. to 11 a.m., thus:

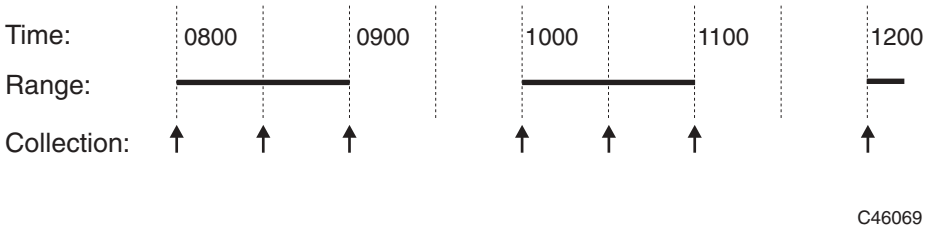


Figure 2-3. *Time ranges*

If you do not specify time ranges, Reporter collects data at the specified interval all day. If you do specify time ranges, Reporter collects data at the beginning and end of each time range and at the end of each interval during the time ranges, but not outside the ranges.

Controlling Intervals

If you were to examine total statistics over relatively long periods of time (such as several hours) less detail would be apparent. To make the data from the subsystems meaningful, you divide the collection periods into equal *intervals* of data collection. The collection interval is the time between *collection instances*.

You can specify the length of the collection intervals. For example, you can specify that the time range from 8 a.m. to 10 a.m. is to be broken into 15-minute intervals. Data would be collected at 0, 15, 30, and 45 minutes past each hour. Statistics would be collected at the beginning of the first interval and at the end of each interval, as shown in Figure 2-4.

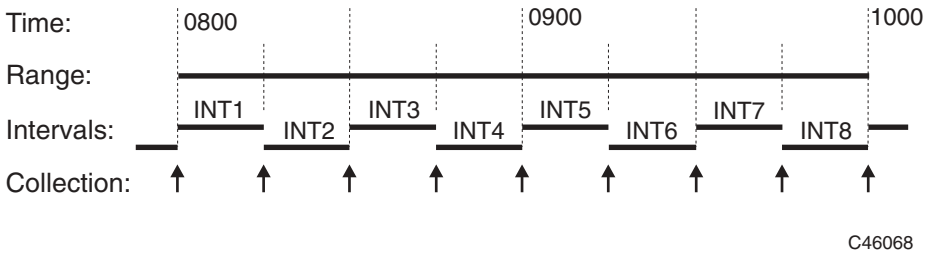


Figure 2-4. *Times of collection instances*

In this example, interval 1 lasts from 8 a.m. to 8:15, interval 2 is from 8:15 to 8:30, and so on. There are 8 equal intervals. If collection is to start at 8:00 a.m., statistics are transferred from the subsystem to Reporter at 8:00 a.m. (taking a base reading), and at 8:15, 8:30, 8:45, and so on.

Synchronizing Collection

You can *synchronize* data collection so that one of your intervals starts at a particular number of minutes after the hour. You can specify that collection intervals are to start at times that are synchronized with RMF data collection for your installation.

You can also specify that there is to be no synchronization, in which case collection is synchronized with the beginning of the collection range, if you have specified a range. However, if you specify neither ranges nor synchronization, SVAA starts collecting immediately and continues collecting until stopped (for performance-tracking data collection) or until midnight (for on-request data collection).

No matter what you specify for synchronization, there are always collection instances at the beginning and at the end of each time range (unless you are collecting subsystem data all day).

Specifying a synchronization point has the effect of moving the beginning and ending points of intervals to coincide with the synchronization time. The length of the intervals then determines the other times that collection intervals are to begin. For example, if you specify a time range from 8 a.m. to 10 a.m., intervals of 20 minutes, and a synchronization time of 5 minutes past the hour, data is collected as shown in the following diagram.

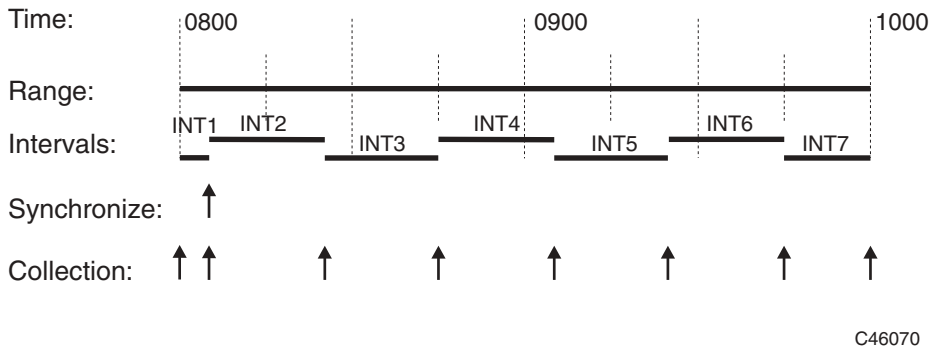


Figure 2-5. *Synchronization out of step with intervals*

In this example, there is a collection instance at 8:00 (taking the base reading) and again at 8:05. Interval 2 ends and collection occurs at 8:25, interval 3 ends and collection occurs at 8:45, and so on. Interval 7 ends at 10:00 a.m. and collection occurs at that time. Notice that that there is data collection at 8:00 a.m. and at 8:05, and that interval 1 is only 5 minutes long. Also, interval 7 is shorter than the other full intervals by 5 minutes. In this example, specifying synchronization points of 5, 25, and 45 minutes would all have the same effect.

Such a combination of intervals and synchronization times as in this example is not recommended. To avoid having irregular collection periods, which could skew the data you obtain, you should specify that the time range would be from 8:05 until 10:05. Alternatively, you should specify that synchronization begin at 0 minutes past the hour:

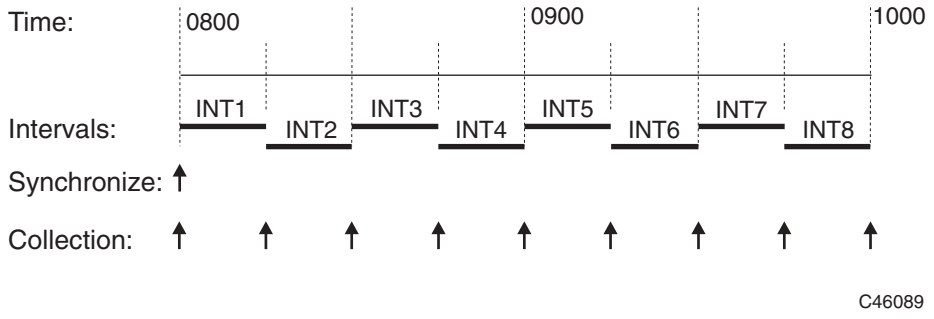


Figure 2-6. *Synchronization in step with intervals*

Issuing a command to collect data during a time range can also cause irregular first intervals to occur. For example, if you start OR collection at 8:50 when the time range is 8:00 to 10:00 and the collection interval is 15 minutes (collection instances occur every 15 minutes), and synchronization is on the hour, collection occurs at 8:50, 9:00, 9:15, and so on, so the first collection interval is shorter than the others:

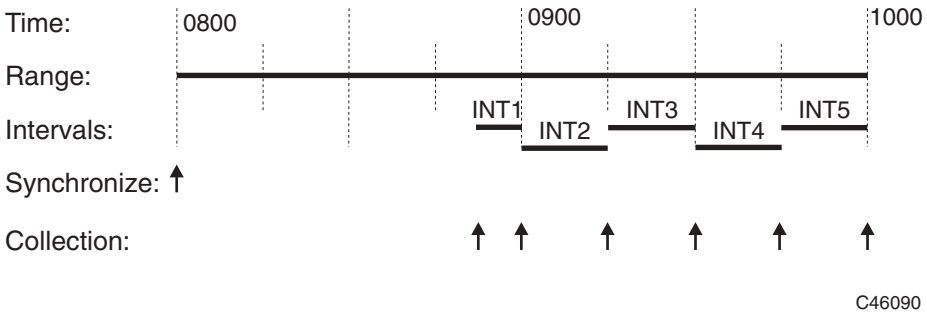


Figure 2-7. *Irregular first interval*

Controlling the Duration of OR Collection

For OR collection only, instead of specifying a time range you can alternatively specify a *duration* of collection. Collection starts at the time you enter the collection command and continues for a duration you specify. A duration is essentially a single time range.

You can use intervals, time synchronization, and RMF synchronization in connection with a duration, just as you can use them with time ranges. However, you can specify only one duration for collection, in contrast to time ranges, of which you can specify up to 16. For example, suppose you start OR collection at 8:30 a.m., you want collection to last for 45 minutes, and the collection interval is 15 minutes:

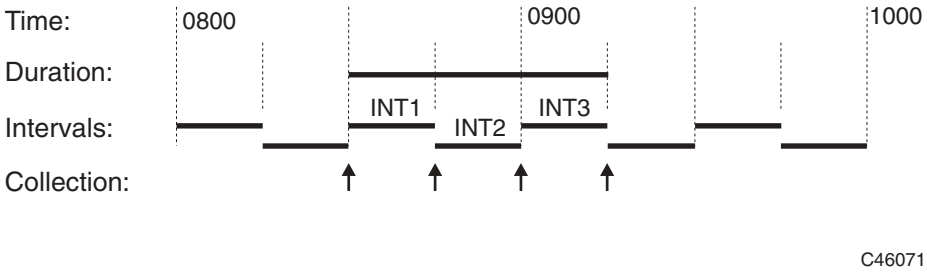


Figure 2-8. *Specifying a duration*

The OR collection interval lasts from 8:30 to 9:15. Reporter reads the statistics from the subsystem at 8:30, 8:45, 9:00 and 9:15, and creates three records, each spanning 15 minutes.

Controlling the Devices to Collect Data From

You can specify either subsystems or specific devices within subsystems for data collection. You can list either unit addresses or volume serial numbers (volsers) for the devices, or both. Reporter lets you specify exactly which devices to *include* in data collection. However, there are many variations among these arrangements.

Instead of specifying lists of functional devices for collection, you can specify ranges of SVAA subsystem names, volsers, or unit addresses. In addition, you can specify certain devices or ranges of devices to *exclude* from data collection. With exclusion, you subtract devices from the list already specified for inclusion in collection. You can even use wildcard characters to specify devices or ranges of devices for inclusion or exclusion.

Reporter thus provides great flexibility in enabling you to specify the devices for data collection. You can also specify collection from non-SVA DASD, and you can collect data for the SVA Test or Production partitions (with overall totals), or you can collect detailed data for each functional device. For non-SVA DASD, Reporter

collects the same cache-effectiveness statistics as for SVA DASD, except for the low reference count and track occupancy statistics—see page 2-3. Reporter does not collect space-utilization data for non-SVA subsystems.

Notes:

1. Performance data is not collected for Power PPRC Bridge devices.
2. In the SAS Macro library provided with SVAA, Power PPRC Bridge devices are considered CKD devices for the purposes of space-utilization data collection and reporting.
3. PAV Alias devices are excluded from all data collection; thus, performance data is not reported for Aliases.

Multiple Collection Tasks and Subsystems

One Data Collection Task can collect data for multiple DASD subsystems. However, to avoid redundant data, you should not have multiple Data Collection Tasks collecting data for the same subsystem.

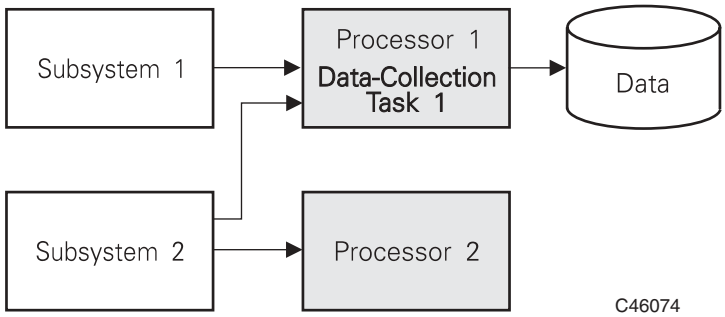


Figure 2-9. *One Data Collection Task for two subsystems.* Processors 1 and 2 can also be logical partitions of a single processor.

For example, in Figure 2-9, Processor 1 has access to both Subsystem 1 and Subsystem 2; Processor 2 has access only to Subsystem 2. For this configuration, you can define one Data Collection Task to run on Processor 1 and collect data for both Subsystems 1 and 2. In this case, Processor 2 does not need to run a Data Collection Task.

Figure 2-10 shows the addition of Subsystem 3 to the configuration.

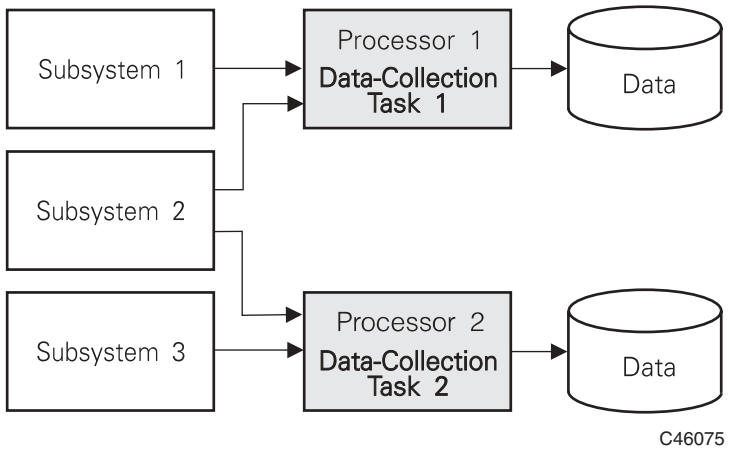


Figure 2-10. *Two Data Collection Tasks for three subsystems*

You can still run one task on Processor 1 to collect data for Subsystems 1 and 2. However, you should now define a Data Collection Task to run on Processor 2 to collect data for Subsystem 3, because the Data Collection Task in Processor 1 does not have access to Subsystem 3.

Note: In Figure 2-10 on page 2-8, a Data Collection Task 2 running in Processor 2 would collect information about Subsystem 2 that is redundant with the data collected in Data Collection Task 1. You should choose to exclude Subsystem 2 from collection in Data Collection Task 2.

With SVAA, you can combine the data from each Data Collection Task into a single file for combined reporting for all your subsystems.

Managing the Collected Data

There are three tasks involved with managing the collected data:

- Copying the logging files, usually at the end of the day.
- Copying and switching the logging files if they become full.
- Summarizing the collected data

Chapter 5, “Managing the Collected Data” provides details of specifying data management.

Copying the Logging Files

Depending on how you use and name the logging files, you may want to start with fresh copies of the logging files each day, so that each day's files are uniquely identifiable.

Handling Full Logging Files

Unless you are using SMF, as the logging files become full, you copy the data to other files that you will use later as input to the summarization step. On-request collection is usually for a relatively short time, so the data sets can be made large enough so that there is no need for copying.

Summarizing the Data

As already described, Reporter stores the data collected from the OS/390 subsystem in one of two ways, at your option:

- With your system SMF data
- In sequential logging files

You can use Reporter to summarize the collected data (from either SMF or the logging files) for input to SAS, which produces reports printed in a columnar format, or to SAS/GRAPH for a graphic report.[1] SVAA also produces a SAS program that can be run later to summarize the data. Reporter stores the summarized data in a file called the *history file*.

Data from previous history files can be combined with the current logging data to produce a new history file.

[1] SAS and SAS/GRAPH are separate products sold by SAS Institute Inc.

Controlling the Files to be Summarized

You can specify as input files to be summarized either your accumulated records in SMF or any number of previously unsummarized logging files. You cannot use a combination of these sources of input. Reporter merges these input files with the previous history file to create a new history file.

Controlling the Output from Summarization

If you plan to use the SVAA summarization function, you must specify the name of a SAS data set for the new history file.

Controlling the Summarization Interval

If you plan to use the SVAA summarization function, you can specify the period of time for which the input data is to be summarized. For example, the collection interval was 15 minutes (that is, collection instances occurred every 15 minutes), but you decide that the collection data should be summarized in two-hour periods. (The summarization interval must be a multiple of the collection interval.)

Controlling the Data to be Retained

If you are working with the previous history file as input, you have the option to specify how many days' data is to be kept in the new history file. This enables you to limit the amount of data in the history file to the number of days for which you intend to produce reports. The default is to keep 13 months' data, which are usually sufficient.

Reporting on the Data

Reporter provides the following reports, which are based on the summarized data described above. You can request:

- Detailed and/or summary reports for specific devices, for a specific date or time interval, or for ranges of dates or time intervals
- Daily, weekly, or monthly summary reports
- Reports showing statistics in bar-chart (histogram) form
- Reports that flag the data when the subsystem statistics exceed or fall below the thresholds that you specify
- Files that can be used as input to the SVAA sample graphics program or to your own graphics programs

Reporter also produces Space Utilization reports on the amounts of functional space currently defined, allocated, and in use at an instant in time.

You control the scope of these reports. For example, you can specify which functional devices are to have their statistics reported. You see the data as computed averages, percentages, and totals.

You can also produce reports from the summarized data using your own report writer or graphics display program.

Chapter 6, “Controlling the Reports” describes how to specify the reports you want Reporter to produce.

Selecting Types of Reports

You can select the Space Utilization reports or various types of Cache Effectiveness or Device Performance reports.

Space Utilization Reports

A Space Utilization report provides a view of conditions in the subsystems at a particular time. The report shows the amount of functional space defined, allocated, and in use, as well as the amount of physical back-end space in use by each functional device.

Device Performance and Cache Effectiveness Reports

Device Performance and Cache Effectiveness reports are based on statistics collected over a period of time from the subsystems. There are overall summary, interval, time-ordered, and daily, weekly, and monthly summary, and exception reports for both device performance and for cache-effectiveness.

Overall Summary Report

An overall summary report produces summaries for the entire report period by subsystem, by individual functional device, and by reporting period. Reporter produces a summary report for each time range you specify.

A summary appears at the end of each overall summary report. The summary shows the totals by Test partition, Production partition, and overall subsystem.

Interval Report

An interval report provides the statistics by subsystem within each interval. You can specify that the reports show statistics for subsystem totals only, or you can also have the report show details for each functional device. Thus, if you have two subsystems, Reporter produces two reports, one for each subsystem, each report covering each specified interval.

Time-Ordered Report

A time-ordered report provides statistics by interval within each subsystem. You can specify that the report show statistics only for each partition and for subsystem totals, or you can also have the reports show details for each functional device. These reports provide the same statistics as the interval report, except that the statistics are organized for each interval, rather than by subsystem.

Summary Reports (Daily, Weekly, or Monthly)

Daily, weekly, and monthly summary reports summarize statistics for each time range. You can specify that the report show statistics only for each partition and the subsystem totals, or that the reports also show details for each functional device.

A daily summary report is the same as an interval report for one day with each of the reporting intervals in that day specified. A weekly summary provides the totals for a week, with separate statistics for each interval. A monthly summary provides the totals for a month, with separate statistics for each interval. There is no summary of all the intervals taken together.

Each entry in a weekly summary report summarizes statistics for Sunday through Saturday of one week. Each entry in a monthly summary report summarizes statistics from the first day of a calendar month to the last day of that month.

A summary appears at the end of each overall summary report. The summary shows the totals by Test partition, Production partition, and overall subsystem.

Exception Report

An exception report for an SVA device shows statistics that fall below or exceed certain threshold values during the times covered by the report. You can specify that the reports show statistics for each partition and for subsystem totals, or you can also have the reports show details for each functional device. You can specify the exception thresholds or take the defaults; Reporter flags the values that go beyond the threshold values for each device.

Bar Chart

A bar-chart (histogram) report represents selected values from one or more of the other types of report, plotted over time.

Graphics

Users with SAS/GRAPH installed can request graphic displays of interval, time-ordered, daily, weekly, or monthly summary report data. Reporter saves the summarized data from the report in SAS data set format to be sent to a separate SAS/GRAPH program. Reporter includes sample SAS/GRAPH programs for producing graphics output from the saved data. Input data to SAS/GRAPH must already be summarized; no further SAS computation is necessary. SAS/GRAPH output includes trend analysis.

Controlling Report Input

Input can be either from the SAS history file, directly from SMF, or directly from the logging files. You cannot use a combination of these sources of input.

Controlling the Time Ranges for Reporting

For reporting, you control time ranges, intervals, synchronization, and durations as you do for data collection. Each time range you specify for reporting must be a multiple of the range for the corresponding collection run. For example, with half-hour collection periods, you should not specify 45-minute reporting ranges, but you can specify one-hour reporting ranges, as shown in Figure 2-11.

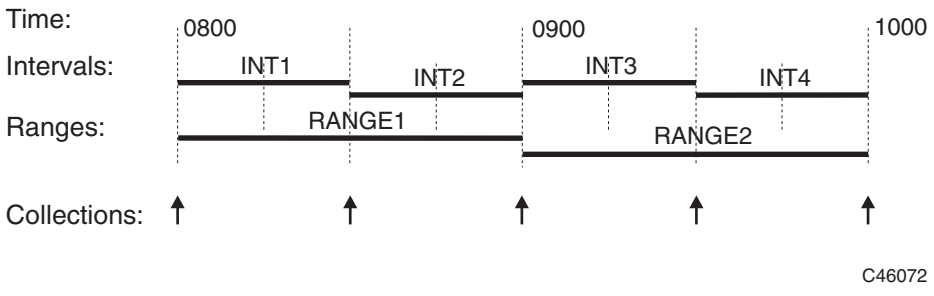


Figure 2-11. *Time ranges for reporting*

Note: In SVAA and RMF reports, the time indicated for an interval is the *ending* time. However, in MICS reports, the indicated time is the *starting* time of the interval.

Specifying Days of the Week to be Included

You can specify the days of the week that are to be included in or excluded from reporting.

Specifying Dates to be Included

You can specify the individual dates or ranges of dates that are to be included in or excluded from reporting. Reporting for holidays, for instance, can be controlled.

Specifying Exception Thresholds

For most statistics that you gather from an SVA subsystem, you can specify a report of those values that exceed or fall below a value you choose or that fall within or outside a specified range of values. These values are called *exception thresholds*.

Additional Software

To use Reporter, you may need the following additional software:

- SAS, if you want to use SAS to produce reports or to summarize the data with Reporter.
- SAS/GRAPH, if you want to produce graphic displays of the reported data.
- ISPF, if you want to use the interactive panel interface to Reporter.
- TSO/E, if you want to use REXX EXECs in OS/390 or if you want to use the interactive panel interface to Reporter.
- An access authorization program such as RACF, if you want to verify a user's authority to use certain Reporter commands.

Note: For the specific versions or releases of these programs required for use with this release of SVAA, see the *SVAA for OS/390 Installation, Customization, and Maintenance* document.

Chapter 3. Reporter in the OS/390 Environment

Chapter Summary

Subsystem Reporting

Figure 3-1 illustrates the OS/390 environment of Reporter.

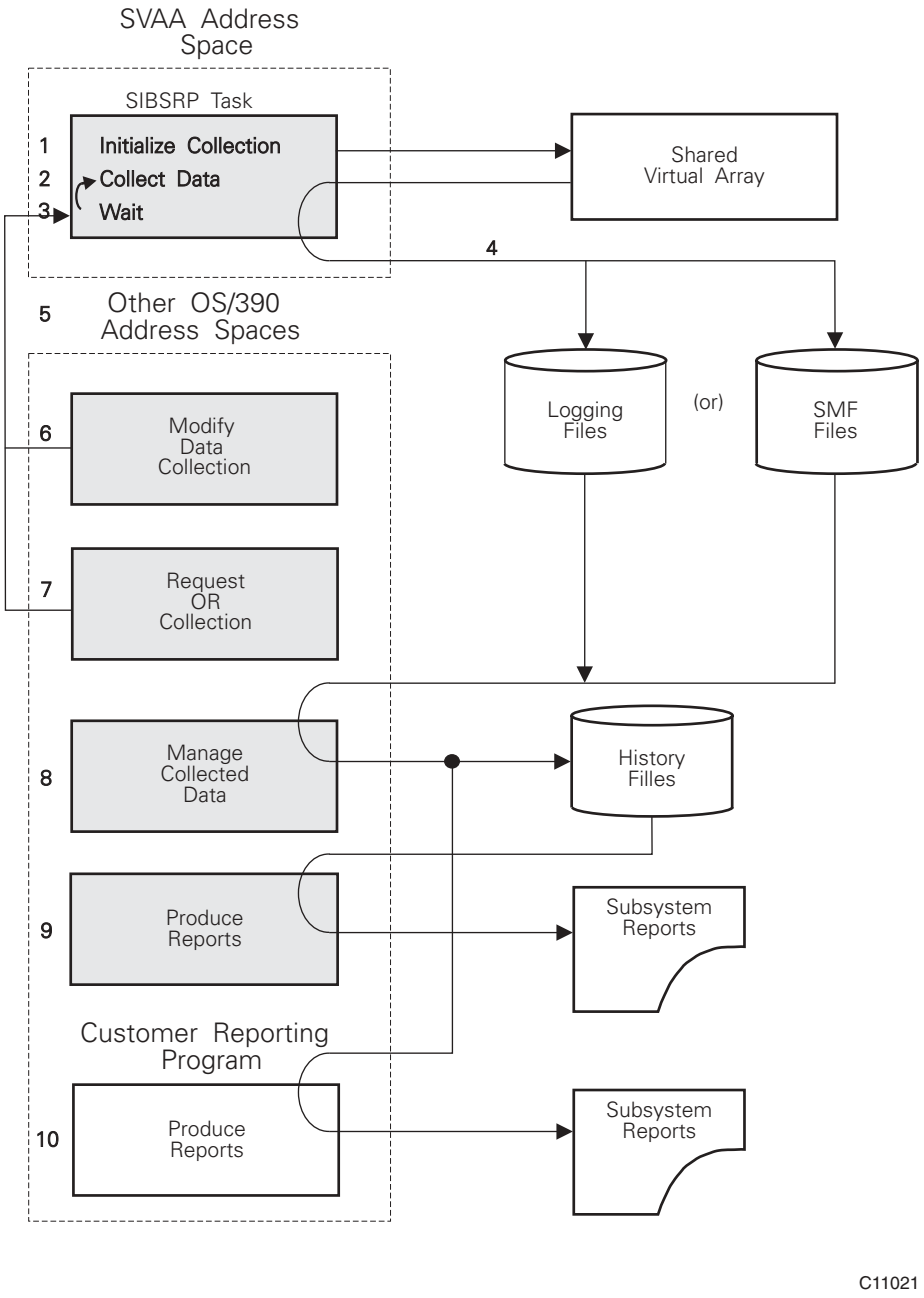


Figure 3-1. *Environment of Reporter*

In the following description of how Reporter operates, the step numbers at the left correspond to the small numerals in Figure 3-1.

1. **Initialize collection.** Run the JCL to start the SVAA address space (you should only run one subsystem). For a sample of this JCL, see “Starting the Data Collection Task” on page A-2. Within this address space, SVAA starts subtasks, including one called SIBSRP. SIBSRP is the Reporter Data Collection Task, which first executes the subcommands contained in an initialization routine found in the SIBSRP*xx* member of SYS1.PARMLIB. These subcommands could include:

- SET ECAMDEVICE subcommands to define the SVA devices that are used for communicating the SVA collection statistics to Reporter.
- SET NONICEBERG subcommands to specify devices that are to be used for communicating non-SVA collection statistics to Reporter.
- An INITIALIZE MAINLOG subcommand to define the files to be used for data collection.
- COLLECT PTDATA subcommands to start performance-tracking (PT) data collection, as defined below.
- COLLECT ORDATA subcommands to start on-request (OR) data collection, as defined below.
- Any other initialization subcommands you want to use, such as SET CASE(UPPER) and SET CMDECHO(OFF).

Note: There are examples and additional information about the SIBSRP*xx* member of SYS1.PARMLIB in the *SVAA for OS/390 Installation, Customization, and Maintenance* guide.

2. **Collect data.** After initializing collection and during the specified time ranges for collection, the Reporter Data Collection Task is usually dormant (not processing), but “wakes up” at intervals you specify to collect data.

   Reporter collects performance data for use as input to the default reports and to any reports that you specify. Primary performance data is called *performance-tracking* (PT) data and is always in SMF record format. As with IBM's RMF, the data for periods outside the specified time ranges are not collected.

3. **Wait**. Once the data for an interval has been collected, the Data Collection Task waits until the end of the next collection interval. The Data Collection Task also waits for commands from a Reporter control utility, such as commands to modify data collection. The task runs until it is specifically stopped.

4. **Store data**. Reporter stores the collected PT data in one or two *logging files*. One of these is the *main logging file*. There may also be a second file, the *alternate logging file*, to be used if the main logging file becomes full. The data is sent to the logging files in SMF format. As an alternative to using the logging files, you can specify that the collected data is to be sent to SMF. It is also possible for different collection runs to collect data in both logging files and in SMF.

   Whenever a logging file becomes full, Reporter invokes an exit in which you can direct SVAA to invoke procedures to offload the data and to switch to the alternate logging file, if one is available. (See “Handling Full Logging Files” on page 5-5.) In the SIBSAMP data set, SVAA provides a sample exit program, SIBSRPSW, and sample procedures that you should modify.

5. **Other OS/390 address spaces**. TSO users or batch jobs can invoke control utilities to change data collection, summarize the collected data, and manage reporting. Multiple TSO and batch address spaces can invoke the utility programs at the same time. Think of these utility programs together with the Data Collection Task as making up Reporter.

6. **Modify data collection**. You can modify data collection in two ways:

   a. Issue commands from the operator console to modify collection parameters or to start, stop, suspend or resume data collection.

b. Issue additional collection commands to modify the data-collection parameters set during initialization.

7. **Request OR collection**. At times, you may find that the parameters you used for data collection are not adequate for understanding some aspect of the performance of the subsystem. In that case, you can request a separate collection run to produce *on-request* (OR) data. Such data can be written to the main logging file, or you can define one or two OR output files for the purpose. These are called the *primary OR output file* and the *alternate OR output file*.

   Using a separate file allows you to retrieve data immediately after the end of a data-collection run. The OR output files are most helpful if you choose SMF management for the main logging file, because SMF data usually cannot be accessed immediately. Reporter provides an exit procedure that allows automatic file switching when a logging file becomes full.

8. **Manage collected data**. You also control how Reporter takes the detailed PT data and summarizes it in a file called the *history file*.

9. **Produce reports with Reporter**. You can control parameters of the reports that you produce with Reporter.

10. **Produce reports without Reporter.** If you do not have SAS installed, you can use other utilities to produce reports from the collected data.

Note: Steps 6 through 9 can be performed in any order and at any time.

SVAA Input

In the OS/390 environment, there are two basic ways of entering input to SVAA:

- With the SVAA set of ISPF panels, as defined for your installation.

  Chapter 7, “Using the SVAA Panels” explains how to use the panels.

- With SVAA subcommands.

  Chapter 8, “SVAA Subcommands and Commands” is an alphabetical reference of the SVAA Reporter subcommands and their parameters.

SVAA Subcommands

SVAA includes two utilities through which you enter the SVAA subcommands:

SIBADMIN Under TSO, the SIBADMIN command creates the environment in which you can issue SVAA subcommands.

SIBBATCH The SIBBATCH utility enables you to program a sequence of SVAA subcommands to be run as a batch job.

Note: You cannot issue the SIBADMIN command within a SIBBATCH program—only subcommands of SIBADMIN can be issued there, such as COLLECT ORDATA and REPORT PERFORMANCE.

SVAA subcommands can be issued in several ways:

- From within SVAA initialization routines.
- From TSO:
  - at the **SIB:** prompt (which appears when you enter the **SIBADMIN** command by itself).
  - within a REXX EXEC
  - within a CLIST
- From within a SIBBATCH file.
- On the command line of the SVAA panels (preceded by **SIB**). See "Entering SVAA Subcommands in the Command or Option Field" on page 7-3.

SVAA Initialization Routines

SVAA enables you to specify initialization information that sets or changes the conditions under which the SVAA subcommands are executed.

SIBSTK00

Optionally, you can place initialization information in a parameter library member named SIBSTK00 in the data set pointed to by the STKPARMS DD statement in the JCL (both for SIBADMIN and SIBBATCH). If STKPARMS is not defined in the executing environment, SVAA looks for SIBSTK00 in SYS1.PARMLIB. SVAA processes the commands in SIBSTK00 (or an alternative member) whenever you use the SIBADMIN or SIBBATCH command.

Note: See "Issuing Subcommands from TSO" on page 3-6 or "SIBBATCH" on page 3-7 for information about specifying an alternative SYS1.PARMLIB member for initialization parameters.

The initialization routine should include a SET ECAMDEVICE subcommand to specify the host address of the device in the SVA subsystem that is to be used for ECAM communications. It should also include a SET NONICEBERG subcommand to designate a 3990 device if you want to collect statistics from a non-SVA subsystem.

PROFSIBA

After processing the commands in SIBSTK00, if you are using SIBADMIN or the SVAA panels, SVAA looks for initialization information in an SVAA macro called PROFSIBA. Information in PROFSIBA overrides parameters specified in the SIBSTK00 member of parmlib, except that SET ECAMDEVICE subcommands are cumulative (that is, subcommands in *both* sources are used).

When you are running SVAA under ISPF, you can specify the profile and parmlib names by using the Session Administration option of the SVAA panels (the values you specify for the **Profile** and **Parmlib** options do not take effect until the next time SVAA is initialized).

Figure 3-2 on page 3-6 is an example of REXX statements used to customize PROFSIBA. You can find this example in the PROFSIBA member of SIBSAMP. You may include as many statements as you want in the profile. See Chapter 8, "SVAA Subcommands and Commands" for information about SET subcommands.

```
/* REXX profile macro for SIBADMIN */
address SIB
"set case(mixed)"
"set ecamdev(100)"
exit
```

Figure 3-2. *Sample REXX profile macro for SIBADMIN*

Figure 3-3 is an example of a CLIST for customizing PROFSIBA. You can find this example in the PROFSIBC member of SIBSAMP. You may include as many statements as you want in the CLIST; see Chapter 8, "SVAA Subcommands and Commands" for information about SET subcommands.

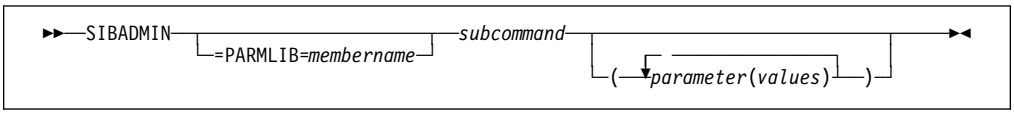
Note: When you copy PROFSIBC from SIBSAMP, rename it **PROFSIBA** before you use it.

```
PROC 1 PROGRAM
CONTROL NOMSG
DATA
SET CASE(U)
SET ECAMDEV(200)
ENDDATA
EXIT
```

Figure 3-3. *Sample CLIST for customizing PROFSIBA*

Issuing Subcommands from TSO

To issue SVAA subcommands from TSO, use the SIBADMIN command. The general form of the command is:



Use the optional **=PARMLIB=** parameter to override SIBSTK00, the default parmlib member that is used in SIBADMIN processing.

A subcommand of SIBADMIN can be:

- Any of the SVAA subcommands described in Chapter 8, "SVAA Subcommands and Commands."
- Any of the SVAA subcommands described in the *SVAA for OS/390 Configuration and Administration* manual.
- A TSO command.

Note: The maximum allowable length of a SIBADMIN command, including its subcommands and parameters, is 950 characters, including blanks.

To process the subcommands, a REXX addressing environment called SIB is established. When running, SVAA displays the **SIB:** prompt, at which you can enter SIBADMIN subcommands. Type **end** to leave SVAA.

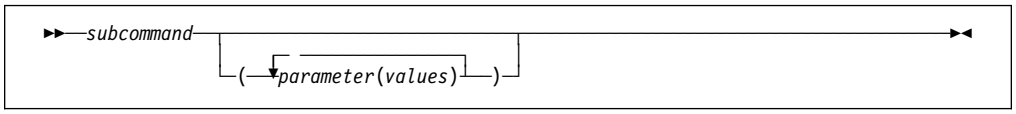
Issuing Subcommands in SVAA Batch Jobs or Macros

Issuing SVAA subcommands in a batch program or a macro is especially effective if you want to issue the same set of subcommands at regular intervals, such as daily or weekly. You can issue the same subcommands in batch jobs or macros that you can issue in SIBADMIN.

In processing subcommands in SIBBATCH or macros, any error that exceeds the value set by the SET MAXRC subcommand causes the program to terminate with an appropriate error message and return code.

SIBBATCH

To run SVAA in batch mode, run the SIBBATCH JCL (shown in Figure 3-4) and include appropriate SYSIN statements containing subcommands of SIBBATCH. The general form of these statements is:



A subcommand of SIBBATCH can be:

- Any of the SVAA subcommands described in Chapter 8, “SVAA Subcommands and Commands.”
- Any of the SVAA subcommands described in the *SVAA for OS/390 Configuration and Administration* manual.

```
// <jobcard>
//STEP1  EXEC  PGM=SIBBATCH
//STEPLIB  DD  DSN=hlq.SIBLOAD,DISP=SHR
//STKPARMS DD  DSN=hlq.sibparm,DISP=SHR
//SIBCTRAN DD  DSN=hlq.SACLINK,DISP=SHR
//CTRANS   DD  DSN=hlq.SACLINK,DISP=SHR
//SYSTERM  DD  SYSOUT=*
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  *
<SVAA subcommands without a preceding SIBADMIN>
   .
   .
   .
/*
//
```

Figure 3-4. *SIBBATCH example*

It is not necessary to provide DCB parameters for the SYSTERM and SYSPRINT data sets. However, if you wish to do so, use:

```
RECFM=VBA,LRECL=137,BLKSIZE=23476
```

The SVAA parameter library contains a member called SIBSTK00, which can specify common startup options, such as **SET ECAMDEVICE**. Startup commands can also appear in the SYSIN data set. If there is no STKPARMS DD statement, but SIBSTK00 appears in SYS1.PARMLIB, SVAA uses the version in SYS1.PARMLIB.

Note: To specify an alternative parmlib member for initialization, use the following EXEC statement in the above JCL:

```
//STEP1  EXEC   PGM=SIBBATCH,PARM='=PARMLIB=membername'
```

Here, *membername*, which can be up to eight characters, overrides SIBSTK00, the default SYS1.PARMLIB member used in SIBBATCH processing.

If the *hlq*.**SACLINK** data set is not in the linklist, use either the **SIBCTRAN** or **CTRANS DD** statement. For an example of the use of **SIBCTRAN**, see the SIBCTUM member of the SIBSAMP library.

Note: If you use a SUMM HIST, REPORT PERF, or REPORT SPACEU subcommand, you must include in the above JCL:

```
//SASPGM   DD  DSN=&&SASPGM,DISP=(MOD,PASS),UNIT=SYSDA,
//             SPACE=(TRK,(5,10)),
//             DCB=BLKSIZE=3120
```

On the **SASPGM DD** statement, specify the appropriate UNIT for the SAS program.

Command Output

The output from the utility commands from a TSO user or batch program (informational messages, error messages, and report output) is handled by default as follows:

- In foreground TSO, SVAA sends the command's message output to the terminal.
- In background TSO, SVAA sends the reports and error messages to the data set named in the SYSTSPRT DD statement.
- In batch, SVAA sends reports and error messages to the data set named in the SYSPRINT DD statement.
- SAS/C writes program output to the data set named in the SYSPRINT DD statement. The statement is optional, but recommended.
- SAS/C writes program error output to the data set named in the SYSTERM DD statement. The statement is optional, but recommended.
- When you start SVAA from the ISPF panels, SVAA sends reports and messages to REXX variables.

You can change the destinations for messages and reports with the OUTM (for messages) and OUTFILE (for reports) parameters of the SET DESTINATION subcommand. However, you cannot change the default destination in ISPF.

Reporter handles data set destinations identified by the OUTMSG and OUTFILE parameters as follows:

- If the data set does not exist, SVAA creates it.
- If the data set exists but has not previously been used in this run of SIBADMIN or SIBBATCH, SVAA replaces the previous content of the data set.
- If the data set exists and has been used previously by an SVAA subcommand in this run of SIBADMIN or SIBBATCH, SVAA appends the new output to the existing content.

You can also redirect the message and report output of a particular subcommand of SIBADMIN. See “Redirecting Subcommand Output” on page 8-10 for more information.

Controlling Reporting Functions

The SIBADMIN command creates the environment in which you can issue Reporter subcommands. There are six SVAA subcommands that control the three functions of Reporter, as follows:

Data collection	INITIALIZE MAINLOG COLLECT PTDATA COLLECT ORDATA
Data maintenance	SUMMARIZE HISTORYFILE
Subsystem reports	REPORT SPACEUTILIZATION REPORT PERFORMANCE

In addition, there are operator-control commands that can modify data collection.

Using System-Operator Commands

You can use commands from the system console to control the Reporter Data Collection Task from another address space. You can also use these commands as subcommands of the SIBADMIN and SIBBATCH commands.

Use these commands to suspend, resume, stop, or display the output of collection runs, or to switch output files between the current logging file and the alternate logging files. You can also use the operator-control commands to modify a collection run that is already running because of a COLL PT subcommand or because of one or more COLL ORD subcommands.

Chapter 8, “SVAA Subcommands and Commands” provides information about Reporter subcommands and their parameters.

Accumulating Data from the Subsystem

At the times set for data collection, Reporter sends a message to the subsystem to obtain the values of the subsystem’s counters. The response is called a *collection instance*. Input counters obtained from the subsystem always consist of raw data. That is, the subsystem does not compute rates or averages.

Reporter maintains accumulators for each data element collected. These collection accumulators contain the current value collected for the data element, which is the value that was collected in the most recent collection instance. Each time a new collection instance occurs, Reporter subtracts the old value from the new value; the difference becomes the computed value for the elapsed interval. The differences between the old values and the current values are sent to the logging files or to SMF.

The logging file always contains the computed values for each element for the elapsed interval. Logging records are time-stamped with the ending dates and times of the data-collection intervals. There is a separate record subtype for each object of data collection: subsystem, channel interface, functional device, and drive module.

See “Space Requirements” on page 5-8 for information about space requirements.

Reporting Information from Collection Runs

Reporter examines the data collected from the subsystems and stored in logging files. You request a performance report based on this data and have a great deal of discretion about the material to be presented in a particular report. For instance, as you become more familiar with SVA performance or as there are changes in the amounts of data or the access density in the SVA, you might change the parameters of exception reporting.

As with data collection, you can control data reporting, either using the Reporter panels or Reporter commands (which can be entered either at the command line or in a batch program). See Chapter 6, “Controlling the Reports” for more information.

Security Checking

Before the Reporter Data Collection Task executes any commands that you send from another source, SVAA verifies your authority to issue the command.

To do this, SVAA calls a user exit called SIBSNDAX. In the SIBSAMP data set, SVAA provides a sample SIBSNDAX module, coded in Assembler Language. You can tailor this module to meet your needs. For example, you can use the IBM Resource Access Control Facility (RACF) or some other security program to specify the list of authorized users.

See the *SVAA for OS/390 Installation, Customization, and Maintenance* manual for more information about security checking.

SVAA Access to Logging Files

Unless you are using SMF for data collection, your site security administrator must grant update authority to the SVAA started task for access to the main logging file and, if one is to be used, to the OR output file.

Chapter 4. Controlling Data Collection

Chapter Summary

Data Collection

This chapter describes how to control data collection with Reporter and deals with the concepts involved with subsystem data collection. Later chapters provide the details of using ISPF panels, system-operator commands, and subcommands of SIBADMIN to control the collection of data.

The following discussion describes each of the collection criteria that are under your control and identifies the subcommand parameters or the panel fields that you use to specify each of the criteria.

There are some data-collection criteria (for example, the destination for the collected data) that you cannot modify after data collection has begun. You can specify them only when you start up data collection. Other criteria can be modified later.

Note: In this chapter, tables show which subcommands and which fields of panels to use to obtain the desired results.

Starting Data Collection

You usually issue a COLL PT subcommand in the initialization statements that Reporter processes when you start SIBSRP. You can issue a COLL ORD subcommand at any time. These subcommands start data collection from your subsystems according to the parameters you issue with the subcommands. There are defaults for most of the parameters you omit.

You can modify data-collection runs that are currently in progress by specifying MODIFY(YES) on a COLL PT or COLL ORD subcommand. If you do so, the parameters on the original subcommand remain in effect unless you override them with parameters on the new command, with one exception. If you originally specified any device-selection parameters, and if you have no device-selection parameters on the new command, the original selection parameters remain in effect. However, if you specify any device-selection parameters on the new command, they completely override and replace the original parameters.

You can start data collection by any of these methods:

- System operator commands. See Chapter 8, “SVAA Subcommands and Commands” for more information.
- Reporter subcommands included in the input PARMLIB statements processed while starting the Data Collection Task. (See Appendix A, “Sample OS/390 Job Control Language” for the JCL used to start up the Data Collection Task.)
- Reporter subcommands from a TSO userid under SIBADMIN.
- Reporter subcommands in a batch program, SIBBATCH.
- Using ISPF panels.

Specifying Where to Send Output

Reporter collects performance-tracking (PT) data in the main logging file and, optionally and if necessary, in the alternate logging file. On-request (OR) data is normally collected in its own files, though the main and alternate logging files can also be used for that data collection, using a unique collection identifier to identify

each collection run. Remember that an OR collection run ends when the specified duration expires.

Before specifying the data-collection options, you must specify the names of the logging files, using parameters of the INIT MAIN subcommand issued in the initialization routine. (There are no corresponding ISPF panel fields.)

You can specify that output is to be sent either to the logging files or to SMF. However, it is possible to specify that the output of the PT collection run be sent to SMF files and that the output of one or more OR collection runs is to be sent to the logging files (see the examples on page 4-4). (You cannot specify that the output of a single collection run be sent to both destinations.)

For information about the amount of storage needed for the collected data, see "Space Requirements" on page 5-8. For sample OS/390 JCL, see "Starting the Data Collection Task" on page A-2.

Note: The logging files have variable-length records. The logical record length (LRECL) must be at least 4092 and the blocksize must exceed the LRECL by 4.

Specifying the Logging Files for PT Data

You can specify one or two files to be used for the logging files for PT data.

To specify the names of the logging files:

	Subcommand	Panel
Name main logging file	OUTF parameter of INIT MAIN	Not specifiable on panels
Name alternate logging file	ALTF parameter of INIT MAIN	Not specifiable on panels

If you use an alternate logging file, use the same form for the filename as the main logging file, but use a different filemode.

Specifying SMF Output

You can specify that collection output is to be sent to SMF rather than to the sequential logging files. Use the SUBPARM statement in the SMFPRMxx member of SYS1.PARMLIB to assign the SMF record number to be used. For information, see the *SVAA for OS/390 Installation, Customization, and Maintenance* manual.

To specify SMF output:

	Subcommand	Panel
Specify output is to SMF	SMF parameter of INIT MAIN	Not specifiable on panels

Specifying the Destination for OR Data

The destination for OR data collection can be either the main logging file or one or two other files that you define (the primary and alternate OR output files). The default is to write the OR collection data in the main logging file.

To specify the destination for OR data collection:

	Subcommand	Panel
Specify primary OR output file	OUTF parameter of COLL ORD	**Primary OR Output File** field
Specify alternate OR output file	ALTF parameter of COLL ORD	**Alternate OR Output File** field
Specify main logging file for OR collection	OUTMN parameter of COLL ORD	**OUTMAIN** field

Identifier for the OR Data-Collection Run

You must specify a unique identifier for an OR collection run.

Note: For a PT data-collection run, you cannot specify an identifier; it is always **PERFORMANCETRACK**.

To specify the OR collection identifier:

	Subcommand	Panel
Name the identifier	COLLID parameter of COLL ORD	**Collection Identifier** field

Example. Directing PT data to SMF and OR data to flat files

In the initialization routine:

```
INIT MAIN (SMF)
```

In SIBBATCH or SIBADMIN:

```
COLL ORD  ( OUTF ( hlq.ORDATA1 ) -
            ALTF ( hlq.ORDATA2 ) )
```

Choosing Data-Collection Options

You can specify the following options for data collection.

Types of Data Collection

You can specify either OR data collection or PT data collection. To specify the type of data collection:

	Subcommand	Panel
Collect OR data	COLL ORD	Reporter main panel, option 2
Collect PT data	COLL PT	Reporter main panel, option 1

Specifying New or Modified Data Collection

For both OR and PT data collection, you can specify that you are defining a new data collection run or you are modifying an existing run.

To specify whether this subcommand specifies a new or modified collection run:

	Subcommand	Panel
Specify modified or new collection run	MODIFY parameter of COLL PT or COLL ORD	**Modify** field of COLLECT PT DATA or COLLECT OR DATA panel

Specifying When to Collect Data

Data collection is the process of reading the accumulators in the subsystem. The times that data collection starts and ends are determined by several interconnecting factors that you specify. The factors controlling data collection are:

- Time ranges. These are the periods of time over which you want to collect data each day. For example:

  ```
  TIMES(1200:1800)
  ```

  specifies that the overall period of time for data collection starts at noon and ends at 6:00 p.m.

  Reporter does not collect data continuously during the time ranges, but at specific times that are determined by the other parameters specified.

- Interval for collection. This determines how often Reporter collects data during the overall collection period. For example:

  ```
  TIMES(1200:1800)
  INT(10M)
  ```

  specifies that Reporter reads the subsystem statistics and writes a record to the logging file every 10 minutes from noon until 6:00 p.m.

- Synchronization. This aligns the first collection interval to a particular minute after the hour. For example:

  ```
  TIMES(1200:1800)
  INT(10M)
  SYNC(5)
  ```

  specifies collection at 12:05, 12:15, 12:25, and so on.

- Duration of collection. For OR collection, this can be used instead of times of collection. For example:

  ```
  DUR(20)
  ```

  specifies that data collection is to start immediately and continue for 20 minutes.

The following sections describe these concepts in more detail.

Specifying Time Ranges for Collection

With times to suspend and resume collection specified, Reporter collects the data without the need for operator intervention. You can specify up to 16 time ranges for data collection to allow automatic suspension and resumption of data collection without stopping the Data Collection Task. This enables you to coordinate collection with your application or business schedules.

If you do not specify any time ranges for data collection, Reporter collects data at intervals over the whole day, at the interval and synchronization you specify.

To specify times for collection:

	Subcommand	Panel
Specify time ranges	TIMES parameter of COLL PT or COLL ORD (see "Specifying Time Ranges for Collection" below)	**Times** field of SPECIFY COLLECTION TIMES panels

The following explanation and examples of time ranges apply to data collection whether you are using either SIBADMIN subcommands or the panels.

Specify a time range in the form *hhmm:hhmm*, where *hh* and *mm* are hours and minutes of the 24-hour clock. The starting time must be between 0000 and 2359; the ending time must be between 0001 and 2400.

Example:
```
0000:2400
0700:0800
1200:1800
```

For OR data collection, collection ranges apply for 24 hours from the time of the COLL ORD subcommand. For any times that fall before the first time that you specify for a range, the corresponding data-collection range occurs on the following day. For example, if at noon you specify:

```
COLL ORD (TIMES(1000:1100 1300:1400 1800:2100) )
```

Reporter collects data from 1 p.m. to 2 p.m., from 6 p.m. to 9 p.m., and from 10 a.m. to 11 a.m. the next day.

Note: If you do not specify any time ranges, OR collection starts at the time you issue the COLLECT subcommand and continues until midnight.

PT collection repeats every day in the specified time ranges. PT collection stops only when the Data Collection Task itself stops, or when you issue a

```
SET COLLECTION (COLLID (PERFORMANCETRACK) STOP)
```

command.

Note: Calculations based on the data that is collected when there is a time change to or from daylight savings time may be skewed because of the time change. You can reduce the effect of the time change as follows:

- Assuming a 15-minute collection interval, setting back the clock at 0200 causes collection instances to have timestamps at 0100, 0115, 0130, 0145, 0100, 0115, 0130, 0145, 0200, and so on. Avoid the duplicate records by issuing a SET COLLECTION SUSPEND subcommand just before the time change, and a SET COLLECTION RESUME subcommand one hour later. SET COLLECTION RESUME causes all the counters to be reset to zero.
- Setting the clock ahead causes a gap in the timestamps of the collection instances, again skewing calculated averages over the affected period of time. Issue the SET COLLECTION SUSPEND and SET COLLECTION RESUME subcommands just before and after the time the clocks are changed.

Follow these rules in specifying time ranges:

- Use leading zeros for hours if appropriate.

  Example: **0700:0800**
  Not allowed: **300:800**[1]

- Starting and stopping times must be different.

  Example: **0700:0800**
  Not allowed: **1100:1100**

- Specify midnight at the beginning of a range as **0000**.

  Example: **0000:0800**
  Not allowed: **2400:1100**

- Specify midnight at the end of a range as **2400**.

  Example: **1800:2400**
  Not allowed: **1800:0000**

- Specify collection for a full day only as **0000:2400**. (This means that you can only specify continuous collection for a full day in a single time range that starts at midnight. However, if you do not specify any time ranges, collection is always continuous.)

  Example: **0000:2400**
  Not allowed: **1200:1200**
  2400:0000

- Use a starting time later than a stopping time to span midnight.

  Example: **2300:1500** (11 p.m. to 3 p.m. next day)

- Separate multiple ranges by spaces.

  Example: **0000:0400 0800:1200**
  Not allowed: **0000:0400,0800:1200**

- Specify time ranges in any order.

  Example: **TIMES(0600:0900 0000:0300 1200:1500)**

- Do not overlap time ranges. (This means that, if you use multiple time ranges, only the range with the latest starting time can span midnight.)

  Example: **TIMES(0600:0900 1200:1800 2100:0300)**
  Not allowed: **TIMES(0600:0900 0800:1000)**

---

[1] However, this is allowed on the ISPF panels; ISPF adds the leading zeros if you omit them.

Example 1. Specifying OR data collection at 1 a.m.

```
TIMES(0400:1200 2000:0200)
```

If you issue the COLL ORD subcommand at 1 a.m. (0100 hours) on December 1, SVAA collects data:

```
0100 to 0200 on December 1
0400 to 1200 on December 1
```

or until collection is stopped, modified, or suspended.

Example 2. Specifying OR data collection at 1 p.m.

```
TIMES(0400:1200 2000:0200)
```

If you issue the COLL ORD subcommand at 1 p.m. (1300 hours) on December 1, SVAA collects data:

```
2000 to 2400 on December 1
0000 to 0200 on December 2
0400 to 1200 on December 2
```

or until collection is stopped, modified, or suspended.

Example 3. Specifying PT data collection at 1 a.m.

```
TIMES(0400:1200 2000:0200)
```

If you issue the COLL PT subcommand at 1 a.m. (0100 hours) on December 1, SVAA collects data:

```
0100 to 0200 on December 1
0400 to 1200 on December 1
2000 to 2400 on December 1
0000 to 0200 on December 2 and succeeding days
0400 to 1200 on December 2 and succeeding days
2000 to 2400 on December 2 and succeeding days
```

until collection is stopped, modified, or suspended.

Example 4. Specifying PT data collection at 1 p.m.

```
TIMES(0400:1200 2000:0200)
```

If you issue the COLL PT subcommand at 1 p.m. (1300 hours) on December 1, SVAA collects data:

```
2000 to 2400 on December 1
0000 to 0200 on December 2 and succeeding days
0400 to 1200 on December 2 and succeeding days
2000 to 2400 on December 2 and succeeding days
```

until collection is stopped, modified, or suspended.

Times to Stop or Resume Data Collection

To suspend or resume collection:

	Subcommand	Panel
Suspend collection	SET COLLECTION SUSPEND command	Not specifiable on panels

	Subcommand	Panel
Resume collection	SET COLLECTION RESUME command	Not specifiable on panels

These commands can be issued from SIBADMIN, SIBBATCH, or from the operator console.

Data-Collection Interval

The data-collection interval determines the length of time between collection instances. The longer the interval, the less detailed is the data for the interval. However, if you set the interval too small, the data obtained may be difficult to interpret. The default interval is 15 minutes.

To specify the interval for data collection:

	Subcommand	Panel
Specify interval	INT parameter of COLL PT or COLL ORD	**Collection Interval** field of COLLECT OR DATA or COLLECT PT DATA

Synchronizing the Collection Interval

You can also specify the minute (0-59) past the hour on which data collection is to be started. Data is collected at the synchronization time and at multiples of the collection interval added to the synchronization time. For example, if the interval is 15 minutes and the synchronization time is 10, data collection intervals start at 10, 25, 40, and 55 minutes past the hour. If the start of data collection plus the collection interval is later than the next synchronization time, the first collection interval is shortened to end at the next synchronization time. All subsequent intervals are the length you specify.

Suppose, for example, that the collection interval is 15 minutes and the synchronization time is 0 minutes. Data collection occurs at 0, 15, 30, and 45 minutes past the hour. If you made the request at 6:10, data collection starts at 6:10, and the first interval is only five minutes. All subsequent intervals are 15 minutes in duration.

If you request synchronization, the collection interval must divide evenly into 60 (that is, it can be 1, 2, 3, 4, 5, 10, 12, 15, 20, or 30). It can also be a multiple of 60. The default is no synchronization.

You can also request the program to use the RMF synchronization value (see “Synchronization of Collection Interval with RMF” on page 4-10).

If you specify no synchronization, Reporter starts collection as follows:

- If you also specify time ranges, at the beginning of the next time range.
- If you do not also specify time ranges, on the next full minute after you issue the COLLECT command.
- If you specify a duration, at the beginning of the duration.

Note: Each collection instance (with the possible exception of the last) is exactly one collection interval after the previous one.

To specify synchronization:

	Subcommand	Panel
Specify synchronization time	SYNC parameter of COLL PT or COLL ORD	**Synchronization Time** field of COLLECT OR DATA or COLLECT PT DATA

Synchronization of Collection Interval with RMF

Instead of explicitly specifying the length and synchronization of the PT collection interval, you can tell the collection program to use the same values as were chosen by your installation for RMF data collection. SVAA searches for these values in an ERBRMF*nn* member, first in the data sets pointed to by the STKPARMS DD statement; if they are not found there, SVAA searches in SYS1.PARMLIB. This allows you to synchronize SVA subsystem performance statistics with the host system RMF statistics.

To specify synchronization with RMF:

	Subcommand	Panel
Specify RMF synchronization	RMF(*nn*) parameter of the COLL ORD or COLL PT	**Collection Interval RMF** field of COLLECT OR DATA or COLLECT PT DATA

Note: In SVAA and RMF reports, the time indicated for an interval is the *ending* time. However, in MICS reports, the indicated time is the *starting* time of the interval.

Duration of Collection

For OR data collection, you can specify that data collection start immediately and last for a specified time period. The maximum duration that you can specify is 9999 hours (over 400 days).

To specify the duration of data collection:

	Subcommand	Panel
Specify OR collection duration	DUR parameter of COLL ORD	**Duration** field of COLLECT OR DATA

Categories of Data to Collect

You can choose between collection of functional-device performance data, cache-effectiveness data, or both. (The default is to collect data for both categories.)

To specify the collection category:

	Subcommand	Panel
Choose device performance	DEVPERF parameter of COLL PT or COLL ORD	**Device Performance** field of COLLECT ORDATA or COLL PT DATA
Choose cache effectiveness	CAEFF parameter of COLL PT or COLL ORD	**Cache Effectiveness** field of COLLECT OR DATA or COLL PT DATA

Specifying Devices for Data Collection

SVAA uses the inclusion parameters (SUBSYS, VOL, and UNIT) and exclusion parameters (EXSUBSYS, EXVOL, and EXUNIT) you specify to select devices for collection (the same principles apply to specifying devices for reporting):

1. If there are no inclusion or exclusion parameters, all devices are selected.
2. If there are any SUBSYS parameters, all devices in subsystems not specified are excluded from selection (even if the devices are specified in VOL or UNIT parameters).
3. If there are any EXSUBSYS parameters, the devices they specify are excluded from selection.
4. If there are any VOL or UNIT parameters, any devices not specified in both parameters are excluded from selection.
5. If there are any EXVOL or EXUNIT parameters, the devices they specify are excluded from selection.

Notes:

1. Wildcard characters and ranges are allowed in inclusion and exclusion parameters (see page 8-6 for information about wildcard characters and page 8-6 for information about ranges).
2. If there is no NONIB parameter, all non-SVA devices are excluded from selection.
3. Performance data is not collected for Power PPRC Bridge devices.
4. In the SAS Macro library provided with SVAA, Power PPRC Bridge devices are considered CKD devices for the purposes of space-utilization data collection and reporting.
5. PAV Alias devices are excluded from all data collection, so performance and space utilization data is not reported for Aliases.

The examples below are concerned with data collection; the same principles apply to data reporting. All examples apply to the following sample configuration of three SVA subsystems, each with four volumes defined:

```
SUBSYSA1          SUBSYSA2          SUBSYSA3
--------          --------          --------
3A0 PROD33        4A0 WORK01        5A0 PAYR01
3A1 PROD44        4A1 PAYR22        5A1 PAYR02
3A2 TEST01        4A2 PAYR23        5A2 TEST03
3A3 PROD34        4A3 TEST02        5A3 PROD41
```

Example 1. Specifying units

```
UNIT(4A0 4A2 4A3)
```

The three devices listed are selected for collection. All other devices are excluded from data collection.

The asterisks indicate the selected volumes:

```
  SUBSYSA1            SUBSYSA2            SUBSYSA3
  --------            --------            --------
  3A0 PROD33        * 4A0 WORK01          5A0 PAYR01
  3A1 PROD44          4A1 PAYR22          5A1 PAYR02
  3A2 TEST01        * 4A2 PAYR23          5A2 TEST03
  3A3 PROD34        * 4A3 TEST02          5A3 PROD41
```

Example 2. Specifying subsystems and units

```
SUBSYS(SUBSYSA1 SUBSYSA2) UNIT(3A0 3A1 4A0 4A1 5A1)
```

There are both SUBSYS and UNIT parameters. To be selected for collection, devices must be specified in both parameters. Devices **3A0, 3A1, 4A0,** and **4A1** are selected for collection. There is no collection for device **5A1**, though it is specified in a UNIT parameter, because SUBSYSA3 is not in the list of selected subsystems.

The asterisks indicate the selected volumes:

```
  SUBSYSA1            SUBSYSA2            SUBSYSA3
  --------            --------            --------
* 3A0 PROD33        * 4A0 WORK01          5A0 PAYR01
* 3A1 PROD44        * 4A1 PAYR22          5A1 PAYR02
  3A2 TEST01          4A2 PAYR23          5A2 TEST03
  3A3 PROD34          4A3 TEST02          5A3 PROD41
```

Example 3. Specifying subsystems, volumes, and units

```
SUBSYS(SUBSYSA1 SUBSYSA2) VOLUME(PROD* PAYR*) UNIT(4A*)
```

To be selected for data collection when the SUBSYS, VOLUME, and UNIT parameters are all specified, a device must be specified by all three parameters. Only devices **4A1** and **4A2** are selected; all other devices are excluded.

The asterisks indicate the selected volumes:

```
  SUBSYSA1            SUBSYSA2            SUBSYSA3
  --------            --------            --------
  3A0 PROD33          4A0 WORK01          5A0 PAYR01
  3A1 PROD44        * 4A1 PAYR22          5A1 PAYR02
  3A2 TEST01        * 4A2 PAYR23          5A2 TEST03
  3A3 PROD34          4A3 TEST02          5A3 PROD41
```

Example 4. Using a wildcard character

```
VOL(PROD3*) UNIT(3A0)
```

To be selected for collection, a device must have both a unit address of **3A0** and a volume serial number starting with **PROD3**. For the sample configuration, only device **3A0** satisfies both these requirements and is selected for data collection. All other devices are excluded from collection.

The asterisks indicate the selected volumes:

```
  SUBSYSA1            SUBSYSA2            SUBSYSA3
  --------            --------            --------
* 3A0 PROD33          4A0 WORK01          5A0 PAYR01
  3A1 PROD44          4A1 PAYR22          5A1 PAYR02
  3A2 TEST01          4A2 PAYR23          5A2 TEST03
  3A3 PROD34          4A3 TEST02          5A3 PROD41
```

Example 5. Specifying ranges of subsystems

```
SUBSYS(SUBSYSA2:SUBSYSA3)
```

All devices in subsystems named **SUBSYSA2** and **SUBSYSA3** are selected for data collection.

The asterisks indicate the selected volumes:

```
  SUBSYSA1            SUBSYSA2            SUBSYSA3
  --------            --------            --------
  3A0 PROD33        * 4A0 WORK01        * 5A0 PAYR01
  3A1 PROD44        * 4A1 PAYR22        * 5A1 PAYR02
  3A2 TEST01        * 4A2 PAYR23        * 5A2 TEST03
  3A3 PROD34        * 4A3 TEST02        * 5A3 PROD41
```

Example 6. Specifying inclusion and exclusion of subsystems, inclusion of volumes

```
SUBSYS(SUBSYSA*) EXSUBSYS(SUBSYSA2) VOL(PROD*)
```

All devices in **SUBSYSA1** and **SUBSYSA3** and with volume serial number starting with **PROD** are selected. For the sample configuration, devices **3A0, 3A1, 3A3,** and **5A3** are selected. No other devices are selected.

The asterisks indicate the selected volumes:

```
  SUBSYSA1            SUBSYSA2            SUBSYSA3
  --------            --------            --------
* 3A0 PROD33          4A0 WORK01          5A0 PAYR01
* 3A1 PROD44          4A1 PAYR22          5A1 PAYR02
  3A2 TEST01          4A2 PAYR23          5A2 TEST03
* 3A3 PROD34          4A3 TEST02        * 5A3 PROD41
```

Example 7. Specifying inclusion because not excluded

```
EXSUBSYS(SUBSYSA3) EXUNIT(*A3)
```

All devices with device numbers ending in **A3** are excluded from data collection. In addition, all devices in subsystem **SUBSYSA3** are excluded. All other devices are selected, because only exclusion parameters are specified. For the sample configuration, devices **3A0, 3A1, 3A2, 4A0, 4A1,** and **4A2** are selected.

The asterisks indicate the selected volumes:

```
  SUBSYSA1            SUBSYSA2            SUBSYSA3
  --------            --------            --------
* 3A0 PROD33          *4A0 WORK01          5A0 PAYR01
* 3A1 PROD44          *4A1 PAYR22          5A1 PAYR02
* 3A2 TEST01          *4A2 PAYR23          5A2 TEST03
  3A3 PROD34           4A3 TEST02          5A3 PROD41
```

Example 8. Using a wildcard character and a range

```
VOL(PAYR*)  UNIT(5A0:5A3)
```

Data is to be collected for any device having a device number **5A0** through **5A3** as well as having a volume serial number that starts with **PAYR**. For the sample configuration, only **5A0** and **5A1** are selected.

The asterisks in the following table indicate the selected volumes:

```
  SUBSYSA1          SUBSYSA2          SUBSYSA3
  --------          --------          --------
  3A0 PROD33        4A0 WORK01      * 5A0 PAYR01
  3A1 PROD44        4A1 PAYR22      * 5A1 PAYR02
  3A2 TEST01        4A2 PAYR23        5A2 TEST03
  3A3 PROD34        4A3 TEST02        5A3 PROD41
```

Example 9. Modifying a data collection run

The following command defines an on-request data collection run:

```
COLL ORD (COLLID(AMADEUS) INT(30) SUBSYS(SUBSYSA1) VOL(PAYR*))
```

The asterisks in the following table indicate the volumes selected for collection.

```
  SUBSYSA1          SUBSYSA2          SUBSYSA3
  --------          --------          --------
* 3A0 PROD33          4A0 WORK01      * 5A0 PAYR01
* 3A1 PROD44        * 4A1 PAYR22      * 5A1 PAYR02
* 3A2 TEST01        * 4A2 PAYR23        5A2 TEST03
* 3A3 PROD34          4A3 TEST02        5A3 PROD41
```

The following command modifies the collection run:

```
COLL ORD (COLLID(AMADEUS) VOL(TEST*) MODIFY(YES))
```

The asterisks indicate the volumes now selected for collection:

```
  SUBSYSA1          SUBSYSA2          SUBSYSA3
  --------          --------          --------
  3A0 PROD33        4A0 WORK01        5A0 PAYR01
  3A1 PROD44        4A1 PAYR22        5A1 PAYR02
* 3A2 TEST01        4A2 PAYR23      * 5A2 TEST03
  3A3 PROD34      * 4A3 TEST02        5A3 PROD41
```

Notice that the original selection of volumes for collection is completely overridden by this command. If you do not specify any VOLUME, SUBSYSTEM, or UNIT parameters on the modified command, the values of those parameters that you specified on the original command are still in effect.

Note: The absence of other parameters does not mean that the parameters now take their defaults. (For example, the collection interval is still 30 minutes.)

Other Collection Choices

This section describes several other choices you have for data collection.

Multiple Data-Collection Runs

Several different collection runs (one PT data-collection run, one or more OR data collection runs) can be active at any given time. Each collection run is uniquely identified. For PT collection, the identifier, which you can omit, is always "PERFORMANCETRACK"; you must specify some other identifier for OR data-collection runs. All output records are stamped with the collection run identifier.

The shortest of the current collection intervals always determines the “wake up” interval for the Data Collection Task. The collection task recognizes when interval end times coincide and writes output data for all affected collection runs.

Batch Control of Collection

Reporter can start, suspend, resume, display, and stop data-collection runs by system operator commands issued within the SIBADMIN or SIBBATCH function. This is useful where console operator communications are not practical. The SIBADMIN or SIBBATCH command terminates after passing the control commands to the Data Collection Task and obtaining the responses. See Chapter 8, “SVAA Subcommands and Commands” for a description of these commands.

SIBBATCH and SIBADMIN provide security checking to ensure that you are authorized to issue the system operator commands. See “Security Checking” on page 3-10 for details.

Output File Disposition

Reporter closes the OR output files when the end time for collection is reached. The main logging file can only be closed by an explicit SET COLLECTION SWITCH command, or by stopping the OR data-collection run.

When you use the SET COLLECTION SWITCH command, Reporter invokes a user exit that determines whether to actually close the main logging file, to switch to the alternate file, or to start writing again at the beginning of the main logging file. (See “Handling Full Logging Files” on page 5-5 for more information.)

Output File Records

Reporter produces output records at the end of the collection interval for each selected device and subsystem. The output includes data for only those devices selected by the UNIT or EXUNIT, SUBSYS or EXSUBSYS, and VOL or EXVOL parameters of the COLL PT or COLL ORD subcommand. Reporter produces only the selected category of output records (device performance, cache effectiveness, or both).

See Table 5-1 on page 5-2 for the kinds of records produced under various circumstances.

Chapter 5. Managing the Collected Data

Chapter Summary

Managing the Data

Reporter collects statistics from the subsystems and either sends the data to SMF or stores the data in the logging files (the main logging file and the optional alternate logging file, and the optional OR output file or files). The data is stored in the logging files in SMF record format, which can be used as input either by SAS or by user reporting programs.

Figure 5-1 on page 5-3 shows the format of the Reporter data at the various stages of the data collection, management and reporting processes.

There are four kinds of records in the logging files and SMF files:

- Subsystem performance records
- Functional device records
- Channel interface statistics records
- Drive module performance records

The following table shows the circumstances under which Reporter produces these records:

Table 5-1. *Kinds of records collected*

Record type	Requested Collection			
	Device Performance		Cache Effectiveness	
	SVA	Non-SVA[1]	SVA	Non-SVA
Subsystem Performance	Yes	No	Yes	Yes
Channel Interface Statistics	Yes	No	No	No
Functional Device Performance	Yes	No	Yes	Yes
Drive Module Performance	Yes	No	No	No

These records are intermingled in the logging files with variable record lengths. For more information, see Chapter 10, “Format of Logging Records.”

There are three independent situations in which the data in the logging files or on-request (OR) output files need attention:

- The daily time for copying the logging files occurs.
- The detailed information in the logging files must be summarized in the history file to conserve storage space and for use in the reports.
- The main logging files or OR output files become full.

This chapter explains what to do in these circumstances.

[1] Device-performance data is never collected for non-SVA devices.

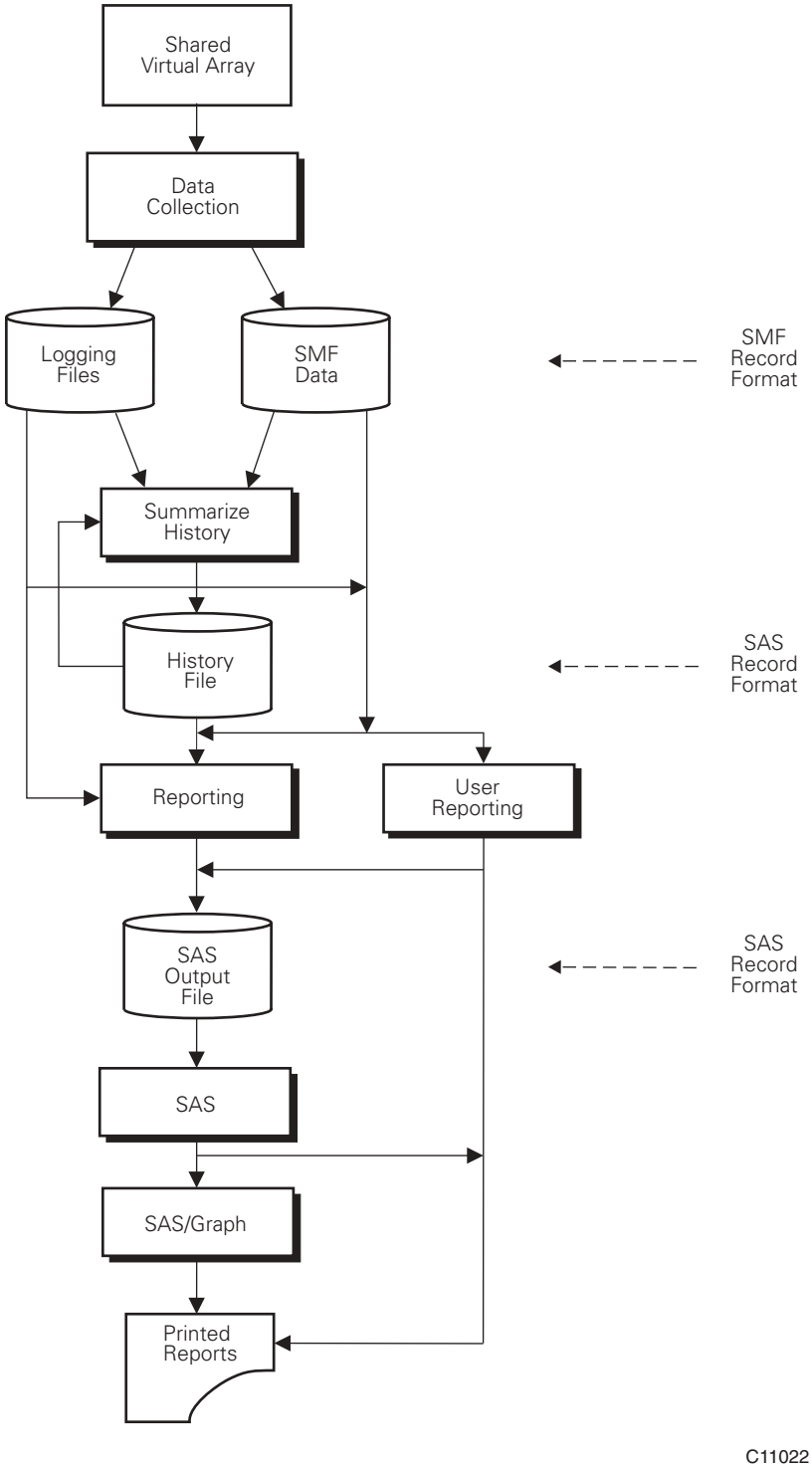


Figure 5-1. *Summary of Reporter data formats*

Copying the Logging Files

At a daily time that you determine, you can run programs that copy the contents of the logging files.

Copy the logging files by modifying and submitting the sample JCL provided in Reporter (see Appendix A, "Sample OS/390 Job Control Language"). You can define these copied files as members of a generation data group (GDG) and maintain them with normal GDG procedures.

Note: The logging files produced by SVAA have variable record format. Do not copy the logging files to fixed-block files.

Summarizing Logged Data

The Reporter SUMM HIST command summarizes performance-tracking data only (even if you use the logging files for collecting OR data, that data is not summarized). There are four kinds of subsystem records in the logging data:

- Subsystem performance records
- Channel interface statistics records
- Functional device performance records
- Drive module performance records

Reporter summarizes the four kinds of records and stores the summarized data in four separate SAS files, one for each kind of record. The four files are collectively called the *history file*. These files will be used later as input to the Reporter reports or as input to the data extraction function of Reporter, which is described later in this chapter.

You can specify up to seven characters for the name of the history file (with the HISTOUT parameter of the SUMM HIST subcommand). In storing the data, Reporter adds a last letter to the filenames of these files to indicate the type of history file:

S Subsystem Performance File
C Channel Interface Statistics File
V Functional Device Performance File
P Drive Module (Physical Device) Performance File

Note: The separate SAS files are in one SAS library (that is, in one OS/390 data set).

After summarization, the history file contains summarized performance-tracking data for the number of days you specify (with the KP parameter of the SUMM HIST command). If you omit the KP parameter, Reporter keeps the data for 395 days (thirteen months).

Note: There is no SVAA panel for the summarization function.

You control the run frequency for summarizing the history file, subject to the following limitations:

- The number of copied files you specify for generation data set retention must be large enough to maintain the detail data for at least the number of days between history file maintenance runs.
- If you direct logging data to SMF, the history file summarization run should be part of the daily or weekly SMF maintenance run.

Reporter summarizes the data over an interval you specify (with the INTERVAL parameter of the SUMM HIST command). Reporter determines the summary interval the first time the history file is produced; it can be specified in subsequent runs but the interval must be a multiple of the original data-collection interval and must be evenly divisible into 24 hours. The larger the summary interval, the less space the history file occupies, but you have less detail. The default summary interval is two hours.

Note: In SVAA reports, the time indicated for the interval is the *ending* time. However, in MICS reports, the indicated time is the *starting* time of the interval.

Reporter stores the summarized data in the history files in SAS format. See Chapter 10, "Format of Logging Records" for the content and layout of the records in these files.

Summarization Process

Reporter can take as input to the summarization process either SMF data or the current members of the copied flat file GDG.

You can write the history file to either tape or disk. A full history file occupies about 600 cylinders of 3380 DASD per subsystem, if you use the default summary interval of two hours and accumulate data for one year.

To summarize the logging data, Reporter does this:

- Obtains the performance-tracking data from the logging files
- Sorts the PT data by subsystem in date and time sequence
- Determines how many day's new data is present
- Makes a new copy of the history file:
  - If you specify an input history file, the data is appended to that file and then summarized. If you specify no such file, the data is summarized and then written to the file specified for history file output.
  - If the file then contains data for more days than you specify with the KEEPDAYS parameter of the SUMM HIST command, Reporter discards the oldest data.

    For example, if you are adding two days' data and are not changing the number of generations of data to keep, Reporter discards the oldest two days' data.
- Uses the output history file you specify (a different name than the input history file)

See "Summarizing the History File" on page A-3 for the JCL to use to summarize the Reporter history file.

Handling Full Logging Files

If a sequential logging file becomes full, Reporter calls a user exit (SIBSRPSW) to determine the action to be taken. (Reporter also calls this exit if you explicitly switch collection to or from the alternate file (with the SET COLLECTION SWITCH command), if there is an open close or access error on the output file, or if there is an I/O error in writing to the file.) The exit can return the name of a procedure to be used to offload the file (that is, to copy the file either from DASD to tape or from

high-performance DASD to low-performance DASD). Reporter includes default file-switching exits for OS/390; you should tailor them for your installation.

The SVAA SIBSAMP data set includes a procedure called SIBOFFLD—sample JCL for copying a full logging file to another data set (for example, a tape or disk GDG). Unless you are using SMF to collect data, you must customize the sample statements for the appropriate file names. (See "Copying Full Logging File Procedure (SIBOFFLD)" on page A-6.)

The file-switching exit (SIBSRPSW) directs Reporter to invoke SIBOFFLD whenever a user logging file becomes full or whenever you explicitly switch to or from the alternate file. The input file is reset to empty once the data has been copied. The SVAA SIBSAMP data set includes a sample SIBSRPSW user exit.

If you choose not to use the SIBSRPSW exit, Reporter takes one of the following actions:

- If a SWITCH command was issued:
  - If there is an alternate logging file, Reporter closes the current logging file and opens the alternate file.
  - If there is no alternate logging file, Reporter ignores the SWITCH command.
- If the current file became full or there was an I/O error, and if there is an alternate file, Reporter switches output to the alternate file.
- If there was an open, close, or allocation error, or if there is no alternate logging file, Reporter closes the output file and terminates all collection runs using the file.

If output is written to SMF, the operating system handles file management.

Note: Even though Reporter automatically switches to the alternate file, the data in the original file is not off-loaded unless the SIBSRPSW exit specifically requests it. Therefore, you should examine the sample user exit, tailor it if necessary, and install the tailored version. You should also tailor and install the sample SIBOFFLD JCL. See the *SVAA for OS/390 Installation, Customization, and Maintenance* manual for more complete information about SIBSRPSW and SIBOFFLD.

Return Codes from SIBSRPSW

Table 5-2 shows the possible return codes from the SIBSRPSW exit.

Table 5-2 (Page 1 of 2). *Possible return codes from SIBSRPSW*

Return code	Reason invoked	Action to be taken by Reporter
0	1, 3	Switch to the other logging file, if it exists, and redrive the I/O
1	1, 3	Switch to the other logging file, if it exists, and bypass the I/O
2	1	Not used in OS/390
3	1	Reposition to the start of the current file and redrive the I/O
4	1	Reposition to the start of the current file and bypass the I/O

Table 5-2 (Page 2 of 2). *Possible return codes from SIBSRPSW*

Return code	Reason invoked	Action to be taken by Reporter
5	1	Close the file and bypass all I/O until the file becomes empty
6	2, 4	Switch to the other logging file and open it
7	2	Ignore the SWITCH command
8	Any	Close the file and stop all collection runs using the file

Extracting History Data for Graphing

SVAA collects from the subsystems statistics that include a large number of variables, resulting in a very large amount of data. SVAA summarizes the data collected for all the reporting variables, as described in "Summarizing Logged Data" on page 5-4. There is an SVAA option, however, that lets you create a SAS program that extracts data from the history file for just the variables you specify. You can reduce the amount of data further by specifying the dates and times, the summarization intervals, and the partitions from which the data you want was collected. You can also specify the type of summary you want, by functional device or by subsystem.

The extracted data is sufficiently moderate in size that you can download the output to a personal computer or a workstation for processing in a report or graph; of course, the data can also be processed on the host. The SAS program created in the process produces data either in a comma-delimited file for processing by a spreadsheet application, such as Microsoft® Excel, or in SAS format for further processing by SAS or SAS/GRAPH.

You can select variables from the categories of channel interfaces, cache effectiveness, or device performance (or any combination of these). A set of panels enables you to choose among the variables to be calculated (those, such as I/O hit percentage, that are calculated from the detail data obtained from the subsystem). These panels also enable you to select identification variables, such as the subsystem name, that help you identify the source of the statistical data. An additional set of panels enables you to specify detail variables, such as functional device utilization time.

SVAA includes in the output data only those variables that you specifically name for extraction. Thus, if you wish to include the detail variables that are involved in a calculated variable, you must specifically name the detail variables to be extracted, as well as naming the calculated variables.

Note: If you request device summarization with variables for both channels and functional devices, SVAA provides only channel totals. (Otherwise, data for a functional device would be collected redundantly over each channel.)

Appendix E, "Using Data Extraction" defines all the calculated and detail variables for which data is summarized by SVAA. The data extraction function is panel-driven only; there are no corresponding SIBADMIN or SIBBATCH subcommands.

The SVAA SIBSAMP data set includes a sample Excel macro and sample SAS/GRAPH programs that you can use to generate graphs from extracted data.

"Executing the Program to Produce Comma-Delimited Output" on page A-7 describes the JCL needed to run the output program.

For information about using the panels for data extraction, see "Extracting Summarized History Data" on page 7-50.

Space Requirements

Reporter requires approximately 320 bytes per collection instance to collect all performance data for a functional device (in the Functional Device Performance File) and about 5200 bytes for the other three records. Data-collection space requirements for a fully populated SVA subsystem (1024 functional devices) can therefore be computed to be:

- About 333K bytes per collection instance
- About 52 cylinders of 3380 DASD per day if you use the default collection interval of 15 minutes
- About 2365 cylinders of 3380 DASD for history data file space per year, if you use the default summary interval of two hours

Assuming 320 bytes per device and 1024 devices per subsystem, each subsystem collection instance produces 81 4-kilobyte blocks of data. If the collection interval is 15 minutes, one SVA subsystem requires 52 cylinders of 3380 DASD data per day. Thus, Reporter needs 520 cylinders of 3380 DASD to maintain 10 days' raw performance data for each subsystem.

Note: For SVA subsystems that support the definition of more than 1024 devices, the above numbers are higher.

For non-SVA subsystems, Reporter requires approximately 268 bytes per collection instance to collect data for a functional device, plus about 192 bytes for the subsystem records. Therefore, the space requirements for a fully populated 3390-3 (64 devices) can be calculated to be:

- About 17.1K bytes per collection instance.
- About 3 cylinders of 3380 DASD per day if you use the default interval of 15 minutes.
- About 120 cylinders of 3380 DASD for history data per year if you use the default summary interval of two hours.

Chapter 6. Controlling the Reports

Chapter Summary

Creating Your Own Reports

You can use the collected data in SMF format and a report-generation program to produce your own reports. In doing so, keep in mind:

- The formats of the four kinds of files in the history file appear in Chapter 10, “Format of Logging Records.”
- During the summarization process, Reporter inserts the current date and time, along with the summarization interval used, in the MEMLABEL field of the SAS variables for the history file.

  Do not modify the MEMLABEL field; it is reserved for Reporter use.

You can run the PROC CONTENTS function in SAS to see the format of the SAS output files.

Requesting Reports

You request reports in one of these ways:

- Using ISPF panels tailored for SVAA. See Chapter 7, “Using the SVAA Panels” for instructions.
- Using the REPORT PERFORMANCE or REPORT SPACEU subcommand, either from the command line or in batch programs. See Chapter 8, “SVAA Subcommands and Commands” for instructions about using the Reporter subcommands to control reporting.

Specifying Report Characteristics

This section identifies the Reporter subcommands and their parameters that control reporting, as well as the panel fields that do the same.

Choosing Report Category

Reporter provides two categories of reports:

- Performance reports. There are two kinds of performance reports:
  - Device Performance reports, which provide statistics that are helpful in understanding I/O performance of the functional devices.
  - Cache Effectiveness reports, which help in determining how effectively the cache is being utilized.
- Space Utilization reports, which show functional and physical space usage and capacity.

See Chapter 9, “Understanding the SVAA Reports” for samples of the reports.

To specify the category of report you want:

	Subcommand	Panel
Performance	REPORT PERF subcommand	Option 3 on SUBSYSTEM REPORTING panel
Device performance	DEVPERF parameter of the REPORT PERF subcommand	**Device Performance** field of SPECIFY PERFORMANCE REPORTING panel
Cache effectiveness	CAEFF parameter of the REPORT PERF subcommand	**Cache Effectiveness** field of SPECIFY PERFORMANCE REPORTING panel
Space utilization	REPORT SPACEU subcommand	Option 4 on SUBSYSTEM REPORTING panel

The default for REPORT PERF is to report both performance types. If there is no data in the input file, Reporter produces a report containing only a header and a message that no cache-effectiveness or device-performance data is available.

Choosing Report Type

"Selecting Types of Reports" on page 2-11 describes the types of reports you can produce with Reporter.

The interval summary, overall summary, time-ordered, and exception reports, and the bar chart can be used to depict short-term subsystem performance. The input to these reports is normally in detail logging files, although you can also use the history file.

The daily, weekly, and monthly summary reports and optional SAS/GRAPH output are used for historical trend analysis. The history file is normally used as input to these reports.

Note: A weekly summary report begins on a Sunday and ends on the following Saturday. A monthly summary report covers all the days of a calendar month.

To specify the type of report:

	Subcommand	Panel
Overall	OVERALL parameter of the REPORT PERF subcommand	**Overall** field of SPECIFY PERFORMANCE REPORTING
Interval	INTRPT parameter of the REPORT PERF subcommand	**Interval** field of SPECIFY PERFORMANCE REPORTING
Daily summary	SUMDAY parameter of the REPORT PERF subcommand	**Daily** field of SPECIFY PERFORMANCE REPORTING
Weekly summary	SUMWK parameter of the REPORT PERF subcommand	**Weekly** field of SPECIFY PERFORMANCE REPORTING
Monthly summary	SUMMON parameter of the REPORT PERF subcommand	**Monthly** field of SPECIFY PERFORMANCE REPORTING
Time-ordered	TIMORD parameter of the REPORT PERF subcommand	**Time-Ordered** field of SPECIFY PERFORMANCE REPORTING
Exception	EXCPTN parameter of the REPORT PERF subcommand	**Exception** field of SPECIFY PERFORMANCE REPORTING
Bar chart	BAR parameter of the REPORT PERF subcommand	**Bar Chart** field of SPECIFY PERFORMANCE REPORTING

The default is to produce the overall report.

Specifying Report Input

You can specify to the Reporter performance-reporting function where it is to find the collected data, either in SAS form in the history files or in SMF form in the detail logging files. To include OR data, you specify an OR collection identifier for the data to include in the report.

To specify where Reporter can find input:

	Subcommand	Panel
SAS history files	RPTSAS parameter of the REPORT PERF subcommand	**SAS File Name** field of the SPECIFY PERFORMANCE REPORTING panel and **SAS Library DSN** field of the SPECIFY PERFORMANCE REPORTING panel
SMF or logging files	RPTFL parameter of the REPORT PERF subcommand	**INFILE DSN** field of the SPECIFY PERFORMANCE REPORTING panel
Identify SMF record number	SMFN parameter of the REPORT PERF subcommand	**SVAA SMF Number** field of the SPECIFY PERFORMANCE REPORTING panel
OR collection input	COLLID parameter of the REPORT PERF subcommand	**Collection Identifier** field of the SPECIFY PERFORMANCE REPORTING panel

Specifying Format of Reports

Normally, Reporter produces tabular reports. You also can specify that bar charts, exception reports, time-ordered reports, interval reports, and the daily, weekly, and monthly summary performance reports (that is, all except the space utilization and overall device-performance reports) be sent to a SAS file for later graphics processing in a separate step. You can use the sample SAS/GRAPH programs (included in the SVAA SIBSAMP library) to process the data, or you can create your own SAS/GRAPH programs.

To specify the format of reports:

	Subcommand	Panel
SAS device detail file	SASDFL parameter of the REPORT PERF subcommand	**SAS Library DSN** and **SAS File Name** fields of SPECIFY REPORT OPTIONS panel
SAS subsystems totals file	SASSFL parameter of the REPORT PERF subcommand	**SAS Library DSN** and **SAS File Name** of SPECIFY REPORT OPTIONS panel

Specifying the SAS Program Name

REPORT PERF and REPORT SPACEU always run under SIBBATCH; they produce a SAS program, which SVAA passes to the next step (the actual invocation of SAS). If you use the ISPF panels, SVAA creates the proper JCL; if you submit your own JCL, it must follow the examples in Appendix B.

Specifying Devices for Reporting

You can include in reporting one or a range of subsystems, volumes, or functional devices (units). You can exclude subsystems, volumes or functional devices, singly or by range, from the devices you define to be included.

Note: Before attempting to specify subsystems or functional devices for data reporting or collection, you must understand how Reporter uses your specifications to select subsystems and functional devices for data collection.

To understand this, study "Specifying Devices for Data Collection" on page 4-11 for the interactions between device inclusion and exclusion parameters; study also the examples in that section.

To specify devices for reporting:

	Subcommand	Panel
Include subsystem	SUBSYS parameter of the REPORT PERF or REPORT SPACEU subcommand	**S** beside **Select Subsystems** on SELECT LISTS panel and **I** on the SELECT SUBSYSTEMS panel, followed by subsystem selections.
Exclude subsystem (on panels, devices that are not specifically excluded are included).	EXSUBSYS parameter of the REPORT PERF or REPORT SPACEU subcommand	**S** beside **Select Subsystems** on SELECT LISTS panel and **X** on the SELECT SUBSYSTEMS panel, followed by subsystem selections.
Include functional device	UNIT parameter of the REPORT PERF or REPORT SPACEU subcommand	**S** beside **Specify Device Ranges** on the SELECT LISTS panel and **I** on SELECT DEVICES or SPECIFY DEVICE RANGES panels.
Exclude functional device	EXUNIT parameter of the REPORT PERF or REPORT SPACEU subcommand	**S** beside **Specify Device Ranges** on SELECT LISTS panel and **X** on SELECT DEVICES or SPECIFY DEVICE RANGES panels.
Include volume	VOL parameter of the REPORT PERF or REPORT SPACEU subcommand	**S** beside **Specify Volser Ranges** on SELECT LISTS panel and **I** on SELECT DEVICES or SPECIFY VOLSER RANGES panels.
Exclude volume	EXVOL parameter of the REPORT PERF or REPORT SPACEU subcommand	**S** beside **Specify Volser Ranges** on SELECT LISTS panel and **X** on SELECT DEVICES or SPECIFY VOLSER RANGES panels.
Include non-SVA devices	NONIB parameter of the REPORT PERF subcommand	**I** beside **Selection Mode** and then **S** beside **Non-SVA Subsystems** on the SELECT SUBSYSTEMS panel.
Exclude non-SVA devices	EXNONIB parameter of the REPORT PERF subcommand	**X** beside **Selection Mode** and then **S** beside **Non-SVA Subsystems** on the SELECT SUBSYSTEMS panel.

The default for reporting is for all devices for which data was collected (for performance reports) or for all installed subsystems (for space utilization reports).

Choosing Degree of Detail

This option controls whether the report should include details for the functional devices you specify. You always obtain the subsystem summary.

To specify the degree of detail for reports:

	Subcommand	Panel
Subsystem totals only	DEVDET(NO) parameter of the REPORT PERF subcommand	**Device Details** field of the SPECIFY PERFORMANCE REPORTING panel
Device details	DEVDET(YES) parameter of the REPORT PERF subcommand	**Device Details** field of the SPECIFY PERFORMANCE REPORTING panel

The default is to report Test partition, Production partition, and subsystem totals only.

Choosing Time Ranges for Reporting

You can specify up to 16 pairs of report starting and ending times (time ranges) for all performance reports to coordinate reports with the collected data. If the starting or ending time falls within a collection or summarization interval, the report includes data for the entire interval. Therefore, for most effective reporting, you may want to coordinate reporting ranges with collection ranges.

Note: Due to processing times, the time stamp on a record can be a few seconds after the ending time of a collection range. Similarly, the starting time of a collection range can appear to occur during a previous collection range.

To specify pairs of starting and ending times:

	Subcommand	Panel
Times	TIMES parameter of the REPORT PERF subcommand	Type **0** in the **Times** field of the SPECIFY PERFORMANCE REPORTING panel and fill in the times on the resulting SPECIFY REPORTING TIMES - PERFORMANCE REPORTING panel

The default is to include all collected data.

Note: The discussion and rules in “Specifying Time Ranges for Collection” on page 4-5 apply in general to reporting, with some significant differences:

1. SVAA first considers for reporting data that was collected on all dates specified with the DATES or DAYS parameters. SVAA then selects for reporting from that data all of the data that was collected in the time ranges specified with the TIMES parameter. That is, SVAA does *not* consider the date and time ranges as defining starting and ending date-time pairs; data is selected first for the specified dates then, from that data, data is selected from the specified time ranges.
2. Data for the ending times of time ranges may appear to be missing from reports. This can happens when, for instance, the interval is 30 minutes, and collection occurs on the hour and half-hour. The data collected at midnight may have an internal time-stamp a few seconds after midnight, because of processing times. The last collection instance included in the report would

occur at 11:30 p.m. (to include the midnight collection, you could specify an ending time of 1201 on the time range).

Example 1. Specifying date and time ranges for data that starts at 1 a.m.

```
DATES(01DEC1998:03DEC1998)
TIMES(0400:1200 2000:0200)
```

If you issue the REPORT PERF subcommand for data that starts at 1 a.m. (0100 hours) on December 1, SVAA includes in the report all data collected:

```
0100:0200 on December 1
0400:1200 on December 1
2000:2400 on December 1
0000:0200 on December 2 and 3
0400:1200 on December 2 and 3
2000:2400 on December 2 and 3
```

Example 2. Specifying date and time ranges for data that starts at 1 a.m.

```
DATES(01DEC1998:03DEC1998)
TIMES(0400:1200 2000:0200)
```

If you issue the REPORT PERF subcommand for data that starts at 1 p.m. (1300 hours) on December 1, SVAA includes in the report all data collected:

```
2000:2400 on December 1
0000:0200 on December 2 and 3
0400:1200 on December 2 and 3
2000:2400 on December 2
```

Choosing Days of the Week for Reporting

You can specify that reporting is to cover specific days of the week, weekdays in general, or weekend days. The days of the week you specify apply to all reports, for the date ranges specified by the DTE and EXDTE parameters.

To specify reporting days:

	Subcommand	Panel
Include specific days of week	MON, TUE, ... subparameters of the DAY parameter of the REPORT PERF subcommand	On the SPECIFY DATE RANGES panel, type **I** in the **Selection Mode** field and **S** next to each weekday to be included.
Include all weekdays	WKDY subparameter of DAY parameter of the REPORT PERF subcommand	On the SPECIFY DATE RANGES panel, type **I** in the **Selection Mode** field and **S** next to each weekday.
Include weekend days	WKND subparameter of DAY parameter of the REPORT PERF subcommand	On the SPECIFY DATE RANGES panel, type **I** in the **Selection Mode** field and **S** next to **SAT** and **SUN**.
Exclude specific days of week	EXDAY parameter of the REPORT PERF subcommand	On the SPECIFY DATE RANGES panel, type **X** in the **Selection Mode** field and **S** next to each weekday to be excluded.
Exclude weekdays	WKDY subparameter of EXDAY parameter of the REPORT PERF subcommand	On the SPECIFY DATE RANGES panel, type **X** in the **Selection Mode** field and **S** next to each weekday.

	Subcommand	Panel
Exclude weekend days	WKND subparameter of EXDAY parameter of the REPORT PERF subcommand	On the SPECIFY DATE RANGES panel, type **X** in the **Selection Mode** field and **S** next to **SAT** and **SUN**.

Notes:

1. If you specify WEEKDAYS for exclusion or inclusion, Monday through Friday are the effective days.
2. If you specify WEEKENDS for exclusion or inclusion, Saturday and Sunday are the effective days.
3. The days you specify with the DAY, or EXDAY parameters of the REPORT PERF subcommand or with the DATE RANGES panel apply to all the performance reports.

Choosing Report Dates or Date Ranges

Date-control parameters allow you to limit reporting to the desired dates. To limit reporting to certain days of the week, see “Choosing Days of the Week for Reporting” on page 6-7.

The following parameters can be specified for all performance reports. The default is to include all collected data.

To specify reporting dates:

	Subcommand	Panel
Include single dates	DTE parameter of the REPORT PERF subcommand	On the SPECIFY DATE RANGES panel, specify dates in the **From-Date** fields and type **I** in the **Selection Mode** field.
Include date ranges	DTE parameter of the REPORT PERF subcommand	On the SPECIFY DATE RANGES panel, specify starting dates in the **From-Date** fields, ending dates in the **To-Date** fields, and type **I** in the **Selection Mode** field.
Exclude single dates	DTE parameter of the REPORT PERF subcommand	On the SPECIFY DATE RANGES panel, specify dates in the **From-Date** fields and type **X** in the **Selection Mode** field.
Include date ranges	DTE parameter of the REPORT PERF subcommand	On the SPECIFY DATE RANGES panel, specify starting dates in the **From-Date** fields, ending dates in the **To-Date** fields, and type **X** in the **Selection Mode** field.

Notes:

1. Specify dates in the form *ddmmmyyyy*, where *dd* is the day of the month, *mmm* is the first three letters of the month, and *yyyy* is the year. In subcommands, specify ranges of dates (pairs of dates for starting and stopping report coverage) in the form *date1:date2*, where each *date* is in the form *ddmmmyyyy*.
2. You can specify dates in the future.
3. You can set up a canned report-performance job that excludes all holidays.

4. Do not specify a date range with two equal dates. Ranges must not overlap.
5. You can exclude one or more dates from the report. For example, you can use the EXDATES parameter to exclude holidays from workday reports.
6. You can specify both report exclude dates and report include dates (but not at the same time on panels).
7. You can select reporting of specific dates in a variety of ways:
   - Multiple starting/stopping time or date pairs.
   - Single starting/stopping time or date pair.
   - Single starting/stopping time or date pair with excluded days.
   - Single starting/stopping time or date pair with included days.
   - Any of the above options plus excluded dates. Excluded dates override included days.
8. With the DTE parameter of REPORT PERF or on the SPECIFY DATE RANGES panel, you can specify up to 16 ranges of dates, up to 16 single dates, or a combination of up to 16 dates and ranges of dates.
9. Similarly, with the SPECIFY DAILY REPORT OPTIONS panel (or the corresponding weekly and monthly report panels), you can specify up to 4 ranges. The dates or date ranges you specify with the EXDATES parameter take precedence over the dates you specify with the DATES parameter. This allows you to exclude holidays.
10. The dates or date ranges you specify with the DATES or EXDATES parameters of REPORT PERF or with the DATE RANGES panel apply to all reports. However, the dates or date ranges you specify with the DATES subparameter of the SUMDAY, SUMWK or SUMMON parameter of REPORT PERF or on the SPECIFY SUMMARY REPORT OPTIONS panel apply only to those reports. If you specify neither the DATES subparameter of a summary report parameter nor the DATES parameter, Reporter produces reports for all data in the history file.
11. If you have intervals that span midnight, your reports will include data from the day previous to the first date you specify for reporting. See Example 4 on page 8-59.

Choosing the Reporting Interval

A report summary interval can be specified for the Interval, Time-Ordered, Performance Bar Chart and Exception reports. The report summary interval must be a multiple of the interval present in the input file (the data collection interval or the historical data summary interval). The default is 120 minutes. (For more information, see “Controlling the Time Ranges for Reporting” on page 2-12.)

To specify the reporting interval:

	Subcommand	Panel
Report interval	RPTINT parameter of the REPORT PERF subcommand	**Report Interval** field of the SPECIFY PERFORMANCE REPORTING panel (enter minutes or hours)

The report summary interval is not specified for the daily, weekly, or monthly summary reports, or for the space utilization reports. Data in those reports is summarized by day, week, or month over the entire interval specified by the report starting and ending times.

Suppressing Report Printing

You can specify that printing of a report be suppressed if you want only a file for SAS/GRAPH output. To obtain SAS file output, you must also specify one or both of the SASDFL and SASSFL parameters.

Suppress report printing by:

	Subcommand	Panel
Suppress printing	SUPP parameter of the REPORT PERF subcommand	**Suppress Print** field of the SPECIFY PERFORMANCE REPORTING panel

Specifying Exception Thresholds

For SVA subsystems, you can specify thresholds for certain variables to be tracked for the exception report. The exception report flags with asterisks the values that represent exceptions from the values you specify for the thresholds. The defaults for these threshold values appear in Appendix B, “Exception Thresholds.”

To specify exception reporting:

	Subcommand	Panel
Exception reporting	EXCPTN (YES or NO) parameter of the REPORT PERF subcommand	EXCEPTION field of the SPECIFY PERFORMANCE REPORTING panel
Threshold values	THRSH parameter of the REPORT PERF subcommand	**LT** and **GT** values for each threshold on SPECIFY EXCEPTION THRESHOLDS panel
Threshold values in an exception file	EXCPFL parameter of the REPORT PERF subcommand	**Exception File Name** field on SPECIFY EXCEPTION THRESHOLDS panel
Partition	PART (PROD or TEST or ALL) parameters of the REPORT PERF subcommand	**Enter Partition** field on SPECIFY EXCEPTION THRESHOLDS panel

Specifying Exception Reporting by Subcommands

With the THRSH parameter of REPORT PERF, specify exception thresholds in sets of three positional parameters enclosed in parentheses and separated by blanks.

```
EXCePTioNs (YES THReSHolds((exception operator threshold)...))
```

where:

exception Is a mnemonic for the report variable that you are specifying the exception for. See Appendix B, “Exception Thresholds” for these mnemonics and for the defaults.

operator Is the comparison to be made: either **LT** (less than), **GT** (greater than) or **NO**. **NO** specifies that there is to be no exception flagging for the variable specified by the *exception* variable. (On the panels, a blank line has the same function as **NO**.)

threshold Is the value you are assigning for comparison with the subsystem values. If you specify the threshold value as an asterisk, Reporter uses the default threshold provided for the report variable. If you specify **NO** for the operator, Reporter ignores the threshold value.

If you do not specify the thresholds for a variable, Reporter uses the default threshold values.

For example, (**DEVAVLPCT LT 99**) requests flagging with asterisks of all intervals in which the device was available less than 99% of the time. (**UTILPCT GT ***) requests flagging of all intervals in which device utilization was greater than the default maximum.

To specify more than one threshold value, enclose each specification in its own parentheses. For example:

```
EXCPTN(YES THRSH((DISCTM GT 15) (CHNLRTE GT *) (ACCDEN GT 12)))
```

Specifying Exceptions

The rules for exception specification either in subcommands or in panels are:

- If you specify both an **LT** value and a **GT** value for the same exception variable and if the **LT** value is less than the **GT** value, Reporter flags any occurrence *outside* the range defined by the **LT** and **GT** values. For example, **(UTILPCT LT 5) (UTILPCT GT 40)** directs Reporter to flag all percentages of device utilization that are either less than 5 or greater than 40 (that is, that are outside the range **5≤UTILPCT≤40**).
- If you specify an **LT** value greater than the **GT** value for the same exception variable, Reporter flags any occurrence *within* the range defined by the **LT** and **GT** values. For example, **(DEVAVLPCT LT 100) (DEVAVLPCT GT 0)** directs the Reporter to flag all devices that were available at least some of the time (that is, that were not disabled) and that were less than 100% available.
- If you specify equal **LT** and **GT** values, Reporter flags all occurrences that do not have exactly that value.
- If you specify only the **LT** value or only the **GT** value, Reporter uses only the range determined by that value. For example, **(DISCTM LT 30)** directs Reporter to flag all disconnect times less than 30 (that is, Reporter does not use the default value for the **GT** operator).
- Separate the positional parameters in each set with blanks and enclose each set in parentheses.
- Specify as many THRSH subparameters in the file as you want. Do not specify duplicate THRSH specifications.
- If you have only one threshold, you can omit one pair of parentheses from the format shown in the box on page 6-10. For example:

  ```
  EXCPTN(YES THRSH(DISCTM GT 15))
  ```

Specifying Exception Thresholds in a File

You can define a set of exceptions in a file and then give the filespec of that file to Reporter, either with the EXCPFL parameter of the REPORT PERF subcommand or with the **Exception File Name** field of the SPECIFY EXCEPTION THRESHOLDS panel. The format of each specification in the file is:

```
THReSHolds (exception operator threshold)
or
THReSHolds ((exception operator threshold)...)
```

where:

exception Is a mnemonic for the report variable that you are specifying the exception for. See Appendix B, “Exception Thresholds” for these mnemonics and for the defaults.

operator Is the comparison to be made: either **LT** (less than), **GT** (greater than), or **NO**. **NO** specifies that there is to be no exception reporting for the variable specified by the *exception* variable.

threshold Is the value you are assigning for comparison with the subsystem values. If you specify the threshold value as an asterisk, Reporter uses the default threshold provided for that field.

Note: See the rules for specifying exceptions in “Specifying Exception Reporting by Subcommands” on page 6-10.

Specifying Exception Reporting on ISPF Panels

You can specify the same information about exception reporting on ISPF panels that you can by Reporter subcommands, except that you cannot specify that there is to be no reporting for an exception threshold. In fact, it is easier to use ISPF panels, because you do not have to look up the mnemonics for the variables you want to be reported.

Using the SPECIFY EXCEPTION THRESHOLDS panel, you can enter the thresholds in a file or you can enter the threshold values under the columns headed **LT** and **GT**. To exclude a variable from threshold reporting, leave the corresponding two fields blank.

Effect of SAS/C Rounding

As described above, exception reports indicate values that fall outside specified thresholds by printing an asterisk to the right of the values in the report. For example, if you specify **UTILPCT LT 40** and the calculated value is 41.37, then 41.4 appears in the report.

Note that SAS/C rounds values for printing in reports to the nearest tenth (one decimal place). This rounding can produce unexpected results in a printed report. For example, if you specify **UTILPCT LT 40** and the calculated value for UTILPCT is 39.97, SAS/C rounds this value to 40.0 and adds an asterisk in the report. The 39.97 value is thus flagged as exceeding the threshold, although in fact the calculated value is within the threshold (it is less than 40). The report can also include values of 40.0 without the asterisk.

For a similar reason, SAS/C can report a value of -0.0. In this case, the calculated value may have been a small negative number, such as -0.03, which SAS/C rounds to -0.0.

Specifying Multiple Reports

You can specify a number of reports with a single REPORT PERF subcommand or with the SPECIFY PERFORMANCE REPORTING panel. Individual reports are requested by report-category or report-type statements.

Sample SIBADMIN subcommand and parameters:

```
REPORT PERF (DEVPERF(YES) CAEFF(YES) SUMDAY(YES) SUMWK(YES) )
```

OS/390 Reporting JCL

See Appendix A, “Sample OS/390 Job Control Language” for the job control language for Reporter.

Chapter 7. Using the SVAA Panels

Chapter Summary

This chapter describes how to use SVAA panels to:

- Set session profile options
- Control data collection
- Specify what data is to be presented in the reports
- Control extraction of data from the history file for downloading to workstation or personal computer.

You can also control reporting with interactive subcommands or, in OS/390, with commands issued in batch programs, as explained in Chapter 8, "SVAA Subcommands and Commands." However, it is generally easier to use the panels than the commands, because then there is no need to remember the syntax of the commands.

Using the SVAA Panels

This section describes how to move among the SVAA panels. The SVAA panels use the terms "select" and "specify" as follows:

select Choose one of the options on the menu and type its single-character identifier in the option field

specify Type appropriate values in the fields on the panel

Specifying Options

On some SVAA panels, you can specify **0** in a field to request Reporter to display a panel on which you can specify options, such as times or dates for collection or reporting or options relating to a type of report. When you return to the original panel, **Y** appears in place of the **0**. If you want, you can cancel the options you have specified by changing the **Y** to **N**.

Entering Values on the Panels

Fields on the panels where "**===>**" follows the field name are data-entry fields, which appear in the figures as underscores, (the underscores do not appear on the actual panels). Fields on the panels whose names end with a colon are display-only fields—you cannot enter data in these fields.

This chapter shows sample entries in uppercase. You can also use lowercase or mixed case.

You can use two keys to move around the panels to make selections or to enter data:

Tab Moves the cursor to the next field in which you can enter data.

When an SVAA panel first appears, the cursor is on the first data-entry field. Use the **Tab** key to move to the next entry field.

Enter Verifies that the values you have entered or options you have chosen on the current panel are allowable. Where appropriate, Reporter then displays the next panel.

Choosing from Tables

Some SVAA panels, such as the SELECT SUBSYSTEMS - LIST MAINTENANCE panel, displays tables of subsystem names, device addresses, or volume serial numbers. On these panels, you select the entries on which you want to act by typing **S** beside the entry. (You can press **Enter** at any time to request Reporter to validate your selection and the data supporting it.)

When you issue the **END** or **RETURN** command, the result depends on whether the data behind the entry is valid (for example, whether a device selection list exists for the entry you have selected).

If any of the data is invalid, Reporter cancels the action and returns you to the panel without saving any of your selections. If all the data you enter is valid, SVAA saves the data.

Using the Command or Option Field

In the command or option field, you can type:

- An SVAA panel command, such as **SUBMIT**.
- An option number on a menu.
- A TSO command.
- An SVAA subcommand (see below).
- An ISPF command. Some useful examples are:

  KEYS Enables you to set your own function key definitions.

  PANELID Causes the panel ID to appear in the upper-left corner of each panel during the current session.

  PFSHOW Displays your current function key definitions at the bottom of the panel. **PFSHOW OFF** turns off function key display. **PFSHOW TAILOR** lets you specify the number and arrangement of key definitions displayed.

  See the ISPF documentation for the complete list of these commands and their explanations.

Entering SVAA Subcommands in the Command or Option Field

You can execute an SVAA subcommand from any SVAA panel by using a SIBCMD command (or simply SIB—the short form) followed by an SVAA subcommand, and pressing **Enter**. (See Chapter 8, "SVAA Subcommands and Commands" for detailed descriptions of the SVAA commands.) The response to subcommands entered in the command or option field depends on the number of lines of output:

- A one-line response appears on the current panel.
- A multi-line response appears on a separate browse panel.

Examples:

```
SIB SET ECAMDEV(123)
```

defines a functional device as eligible for privileged ECAM status.

```
SIB SET COLLECTION (COLLID(SPECIAL) SUSPEND)
```

suspends the collection run having the **SPECIAL** identifier.

```
SIB COLL ORD(COLLID(CT220))
```

requests data collection with the default values for all parameters. The collection identifier is to be **CT220**.

Note: If you enter SIBCMD or SIB without any subcommand parameter, SVAA displays a panel with a prompt at which you can enter one or more subcommands.

Using the Panel Commands or Function Keys

When you are using the SVAA panels, there are a number of panel commands that you can assign to function keys.

If there is no definition for a function key, SVAA does not allow you to use that key or the corresponding command on that panel. Unless you have tailored the function key definitions, the general default key definitions for the panel commands are as shown in Table 7-1.

Not all SVAA panel commands work on all panels. For a list of the commands that you can use with the current panel, enter **?** in the command field.

Table 7-1. *Panel commands for Reporter*

Command	Description	Default Key
ADD	Adds the device to the selection list.	F5
CANCEL	Cancel the operation of the current panel and return to the previous panel.	F15
REFRESH	Replace the values you have entered on the current panel with the values that are currently stored in SVAA.	F24
SUBMIT	Executes the function of the panel with the data you have entered on it. This command can generate an SVAA subcommand and send it to the Reporter Data Collection Task, or it can submit the JCL to start reporting.	F6

Using Line Commands

On some SVAA panels that display rows of data, you can enter certain one-letter commands called line commands to the left of any row. The line commands are:

I*nn* Insert *nn* blank lines below the current line
D*nn* Delete *nn* lines starting with the current line
R*nn* Replicate this line *nn* times and insert the replicated lines below the current line

Refreshing the Default Values

On any panel that displays information about SVA devices, you can press F24 to make sure that the information has not been changed since you first displayed the information. (For instance, the name of an SVA subsystem may have been changed from some other task.)

Help

Use the **HELP** command to display online help for any panel. Scroll up or down the help panels to find help for a particular field. Issue the **END** command to return to the panel for which you requested help.

On any help panel, use the **HELP** command for an explanation of the commands you can use to navigate the help panels.

Messages

SVAA displays brief messages in the upper-right corner of the screen. Issue the **HELP** command to display an expanded explanation.

Note: When displaying an error message, SVAA places the cursor under the field in which it finds the error.

File Names

You cannot use style prefixes on the SVAA panels.

SVAA Main Menu

To display the SVAA main menu, enter ISPF and go through the panels to get to SVAA, as defined for your installation.

SVAA displays a copyright panel and then the Shared Virtual Array Administrator main menu, as shown in Figure 7-1.

```
SIBMM00            *** SHARED VIRTUAL ARRAY ADMINISTRATOR ***
OPTION  ===>

   0  SVAA PARMS    - Specify user parameters
   1  CONFIGURATION - Configure and administer SVA subsystems
   2  SELECTION     - Maintain device selection lists
   3  REPORT        - Perform subsystem data collection and reporting
   4  SNAPSHOT      - Snap data set and snap volume
   X  EXIT          - Terminate SVAA processing


Enter END command to terminate SVAA
```

Figure 7-1. *SVAA main menu (SIBMM00)*

This is the first panel you see for all SVAA functions. Select **0, 2** or **3** on the panel to begin one of the activities described in this chapter.

0 **SVAA PARMS.** Set parameters for your SVAA session. See “Session Administration” on page 7-6.

2 **SELECTION**. Create, modify or delete lists of subsystems or devices. See “Maintaining Selection Lists” on page 7-11.

3 **REPORT**. Control data collection and reporting for DASD subsystems. See “Data Collection and Reporting” on page 7-28.

Options **1** and **4** are documented in the *SVAA for OS/390 Configuration and Administration* manual.

Enter **X** or issue the **END** command to terminate SVAA and return to ISPF.

Each section in this chapter has a flow diagram of the panels you use to perform the task being described.

Note: The four-character labels appearing above each box in the flow diagrams are prefixed with **SIB**. The full identifiers appear on the screen when you issue the **PANELID** command. The figure captions for the panel diagrams in this chapter indicate the full identifiers of the panels.

Session Administration

Session administration consists of setting parameters for your SVAA session. Figure 7-2 shows the sequence of panels for SVAA session administration in OS/390 systems.

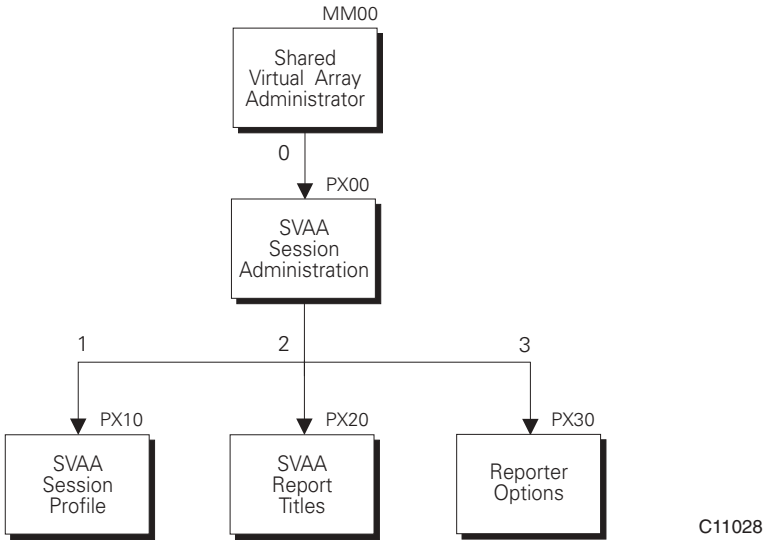


Figure 7-2. *Panels used in session administration*

When you choose option **0** on the SVAA main menu, SVAA displays the SESSION ADMINISTRATION menu, as shown in Figure 7-3. SVAA then displays panels that enable you to set options for your session profile, for report titles, and for your Reporter libraries.

```
SIBPX00           *** SVAA SESSION ADMINISTRATION ***
OPTION ===>

   1  PROFILE       - Specify user profile options
   2  TITLES        - Specify report titles
   3  REPORTER      - Specify Reporter options
```

Figure 7-3. *SVAA Session Administration menu (SIBPX00)*

To set session options, type a number in the OPTION field:

1. Type **1** and press **Enter** to set session profile options

   Result: SVAA displays the SVAA SESSION PROFILE panel. See below.

2. Type **2** and press **Enter** to set report title options

   Result: SVAA displays the SVAA REPORT TITLES panel. See “Specifying Report Titles” on page 7-9.

3. Type **3** and press **Enter** to set subsystem reporting options

   Result: SVAA displays the REPORTER OPTIONS panel. See “Setting Reporter Options” on page 7-10.

Setting Profile Options

The following panel appears when you enter **1** on the SVAA SESSION ADMINISTRATION menu.

```
SIBPX10               *** SVAA SESSION PROFILE ***
COMMAND===>

                  Permanent          Session
Option            Value              Value             Source
--------          ---------          --------          ---------
Case       ===>   MIXED       ===>   MIXED             perm
Pagesize   ===>                ===>   50                profile
Profile    ===>                       PROFSIBA          default

Parmlib    ===>                       SIBSTK00          default
SSName     ===>                ===>   STK1              profile

Job Card
--------
===> //SPKS$    JOB (ACCOUNT),'NAME'
===> //*
===> //*
===> //*
```

Figure 7-4. *SVAA Session Profile panel (SIBPX10)*

This panel enables you to change your profile options—for the current session only (use the **Session Value** fields) or for the current and future sessions as well (use the **Permanent Value** fields).

Changing the permanent values also changes the values for the current session. Changing the session values does not affect the permanent values.

Note: Values you enter on this panel override the corresponding values in your SIBADMIN profile or in the SIBSTK00. member of the parmlib (see “SVAA Initialization Routines” on page 3-5).

The panel displays a table with four columns:

Option The name of the option to be set.

Permanent Value The option value that applies to all sessions. This column is initially blank, although the default values are in effect for all options. Any allowable value you type in this column automatically appears in the **Session Value** column as well, unless you change the session value at the same time. This value applies for the rest of the current session and for all future sessions—until you enter a new value.

Session Value The option value applies to the current session. To set a different value for the current session, type a new value in the field. (After you do so and press **Enter**, the value in the **Source** field changes to **session**.) The new session value does not affect the permanent value.

Source The source of the current value:

- **default** The value is the SVAA default value (which was set during SVAA initialization).
- **session** The value was typed in the **Session Value** column or entered by the appropriate SET subcommand during the current session.
- **perm** The value was typed in the **Permanent Value** column during this session or a previous session.
- **profile** The value was set in the profile macro that was executed when you used SIBADMIN.
- **parmlib** The value was set in a parmlib member.

To specify the profile options:

1. Use the **Case** field to translates values that you enter for parameters of subcommands to uppercase or leaves them in mixed case.

   You can enter either **MIXED** (or **M**), or **UPPER** (or **U**).

2. Use the **Pagesize** field to set the number of lines per page for SVAA reports. You can enter a number from 40 to 99.

3. Use the **Profile** field to specify the SVAA profile to be executed whenever you use the SIBADMIN command in future sessions. (You cannot change the session value for the profile.)

4. Use the **Parmlib** field to specify the parmlib member to be used during initialization of the next SVAA session. (You cannot change the session value for the parmlib.)

5. Use the **SSNname** field to specify the OS/390 subsystem name to be used when another (test) SVAA installation will be in use. Normally, only one SVAA subsystem is active, but sometimes a test subsystem is activated to verify the installation of a new release of SVAA.

6. Use the **Job Card** fields to specify the JCL to be used when SVAA submits a job to OS/390. Tailor the field with your own information (use a $ sign prefixed to your TSO userid).

7. Issue the **END** command to accept the new values and return to the SVAA SESSION ADMINISTRATION menu.

Specifying Report Titles

The following panel appears when you select option **2** on the SVAA SESSION ADMINISTRATION menu and press **Enter**.

```
SIBPX20                  *** SVAA REPORT TITLES ***
COMMAND ===>

SOURCE: Title1- profile    Title2- profile    Title3- profile

SESSION:
Title1   ===> _
         ===>

Title2   ===>
         ===>

Title3   ===>
         ===>

PERMANENT:
Title1   ===>
         ===>

Title2   ===>
         ===>

Title3   ===>
         ===>
```

Figure 7-5. *SVAA Report Titles panel (SIBPX20)*

Use this panel to enter up to three title lines that are to appear at the top center of each page of your reports. The three **SOURCE** fields show the source of the current text for each title. The default is a blank line.

To specify titles for your reports:

1. Use the **Session Title1, Title2**, and **Title3** fields to specify titles for the current session. You can specify up to 127 characters for each title line.
2. Use the **Permanent Title1, Title2** and **Title3** fields to specify titles for future sessions as well as the current session.
3. Issue the **END** command to accept the new values and return to the SESSION ADMINISTRATION menu.

Setting Reporter Options

The following panel appears after you select option **3** on the SVAA SESSION ADMINISTRATION menu.

```
SIBPX30                 *** REPORTER OPTIONS ***
COMMAND===>

SVAA Load Library   ===> hlq.SIBLOAD
SAS/C Load Library  ===>
SVAA SAS Macros     ===> hlq.SIBSAS
SVAA Parm Library   ===>
SVAA Parmlib Member ===>
SAS Proc Name       ===> SAS
SAS MEMSIZE option  ===>        (SAS MEMSIZE option in megabytes)
SAS Program Device  ===>
SAS Work Space
        Unit        ===>        (Type CYL for cylinders, TRK for tracks)
        Primary     ===>
        Secondary   ===>
```

Figure 7-6. *Reporter Options panel (SIBPX30)*

Use this panel to specify the names for SVAA libraries for Reporter.

1. Enter values in the panel fields:

SVAA Load Library (Required) The name assigned to the SVAA load library file during SVAA installation.

SAS/C Load Library (Conditional) The name assigned to the SAS/C load library during SVAA installation. Omit this name if it is specified in the linklist or if you are not using SAS/C.

SVAA SAS Macros (Required) The name assigned to the SVAA SAS macro library file during SVAA installation.

SVAA Parm Library (Optional) The name assigned to the SVAA parameter library file during installation. Default is SYS1.PARMLIB.

SVAA Parmlib Member (Optional) The name of the parmlib member to be used in SIBBATCH processing. Default is SIBSTK00.

SAS Proc Name (Optional) The name of the SAS procedure to be used. Default is SAS.

SAS MEMSIZE option (Optional) The limit on the amount of memory (in megabytes) to be used by SAS. Default is 0.

SAS Program Device (Optional) SAS uses a temporary file specified by the SASPGM DD statement. You can enter a value in this field to override the unit value specified in the JCL. Default is SYSDA.

Unit Enter either **CYL** or **TRK** to specify whether cylinders or tracks are to be used for SAS work space.

Primary The number of units to be allocated for SAS work space.

Secondary	The number of units to be allocated if the primary SAS work space units are exceeded.

2. Issue the **END** command to accept the new values.

Result: SVAA returns you to the SVAA SESSION ADMINISTRATION menu.

Maintaining Selection Lists

Reporter maintains the names of devices in the subsystems in device selection lists. A *selection list* is a named list of one or more:

- Subsystems that are attached to the host in which Reporter is running
- Unit addresses of functional devices within the subsystems
- Volume serial numbers of functional devices within the subsystems

Another form of selection list is a combination of any the above types of selection lists for a single subsystem.

Use these selection lists to specify groups of devices for which you want data to be incorporated in a single report.

When you choose option **2** on the SVAA main menu, Reporter displays the **Device List Maintenance** panel. Reporter then displays a sequence of panels that allows you to add, delete, copy, modify, or browse device selection lists. In addition, whenever you choose an option on one of the SELECT LISTS panels, you can then choose to browse your selection lists.

Figure 7-7 on page 7-12 shows the panels to use to maintain selection lists.

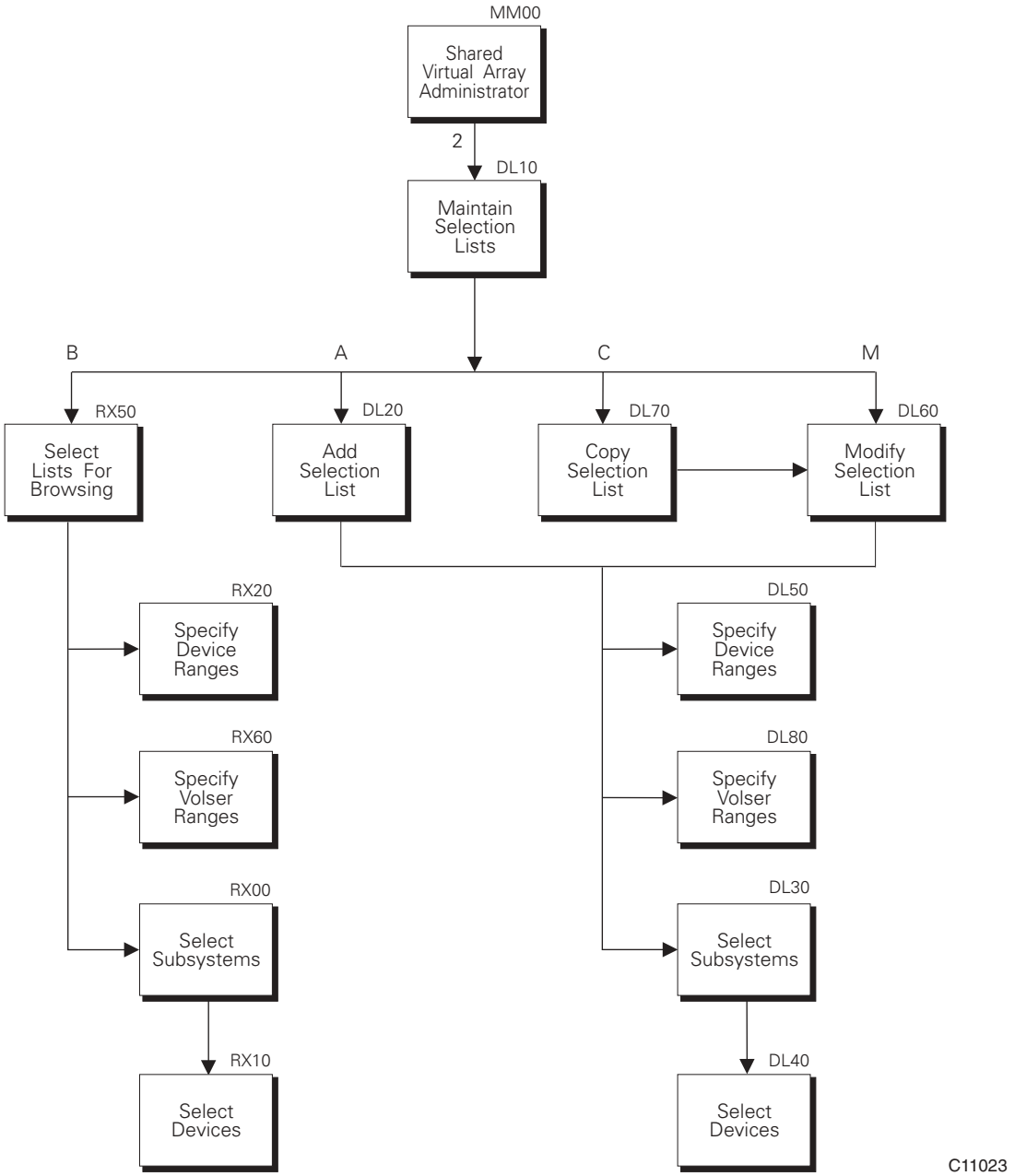


Figure 7-7. *Panels used in maintaining selection lists*

To work with selection lists:

1. Select option **2** on the SVAA main menu to browse, add, copy, delete, or modify a selection list.
2. Press **Enter**.

   Result: The MAINTAIN SELECTION LISTS panel appears, as shown in Figure 7-8 on page 7-13.

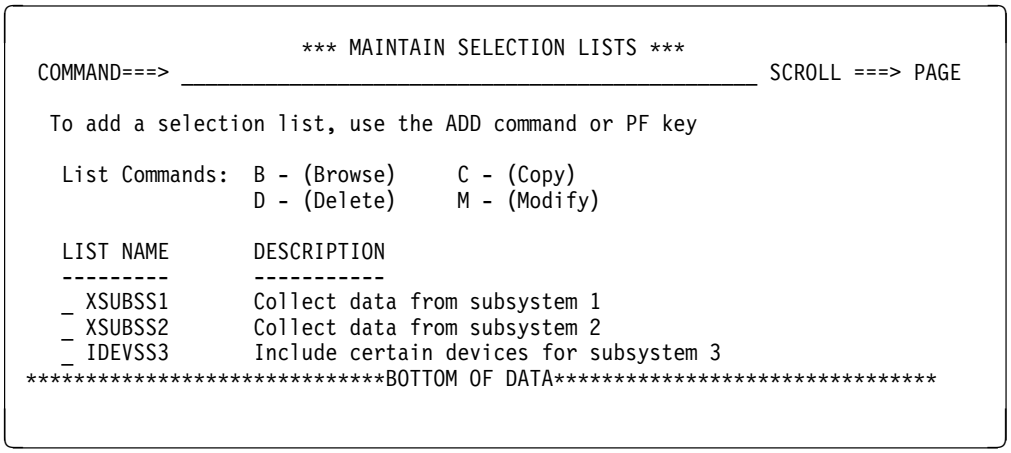


Figure 7-8. *Maintain Selection Lists panel (SIBDL10)*

On this panel, the **LIST NAME** column is a list of names of selection lists. To the left of each name is a field in which you type one of the list commands shown in the panel.

From this panel, you can either add a new selection list, or you can browse, copy, modify, or delete an existing selection list.

3. To add a device selection list, issue the **ADD** command. Go to "Adding a Selection List," below.
4. Type **B, C, M,** or **D** beside the name of a selection list and press **Enter**. The following pages discuss these options.

Adding a Selection List

To add (create) a device selection list:

1. Issue the **ADD** command on the MAINTAIN SELECTION LISTS panel (Figure 7-8).

   Result: The ADD SELECTION LIST panel appears, as shown in Figure 7-9 on page 7-13.

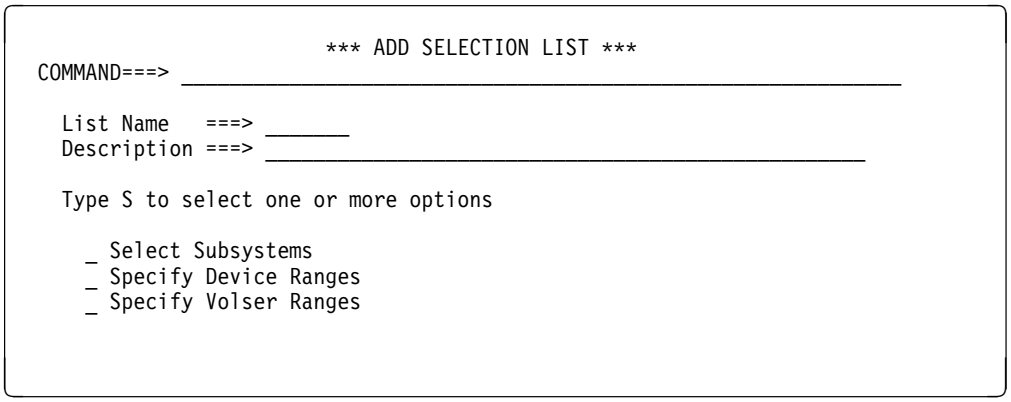


Figure 7-9. *Add Selection List panel (SIBDL20)*

2. Type at the top of the panel up to 7 characters for the name (required) and up to 49 characters for a description (optional) of the selection list you want to add.

Note: List names cannot begin with **SIB**, which is reserved for internal use by SVAA.

3. Type **S** beside any of the types of selection list you are adding (subsystem, device range or volser range, or any combination of these) and then press **Enter**.

   Result: One of the following panels appears, depending on your choice:

SELECT SUBSYSTEMS panel	Go to "Selecting Subsystems and Devices" (below)
SPECIFY DEVICE RANGES panel	Go to "Specifying Device Ranges" on page 7-16
SPECIFY VOLSER RANGES panel	Go to "Specifying Volser Ranges" on page 7-17

   Note: If you select more than one of the options, you see the appropriate panels in turn.

Selecting Subsystems and Devices

The following panel appears when you type **S** beside the **Select Subsystem** field on the ADD SELECTION LIST panel.

```
            *** SELECT SUBSYSTEMS - LIST MAINTENANCE ***
 COMMAND===> _____________________________________________ SCROLL ===> PAGE


  Selection List: XSUBSS1
  Selection Mode ===>  _           I = include, X = exclude

  Type S to select subsystem
  Type L to create device selection list   (SVA subsystems only)
  Type M to modify device selection list   (SVA subsystems only)

    _  Non-SVA Subsystems

  If no subsystems are specifically marked, by default all existing
  SVA subsystems will be included for collection/reporting.

     _  ICESYS00       _  ICESYS01       _  ICESYS02       _  ICESYS03
     _  ICESYS04       _  ICESYS05       _  ICESYS06       _  ICESYS07
     _  ICESYS08       S  ICESYS09 *     _  ICESYS10       _  SVASYS15
     _  SVASYS12       S  SVASYS13 *     _  SVASYS14       _  SVASYS17
     _  SVASYS16       _  SVASYS17       _  SVASYS18       _  SVASYS21
     _  SVASYS20       _  SVASYS21       L  SVASYS22 *     _  SVASYS23
     _  SVASYS24       _  SVASYS25       _  SVASYS26       _  SVASYS29
 ******************************BOTTOM OF DATA*********************************
```

Figure 7-10. *Select Subsystems - List Maintenance panel (SIBDL30)*

On this panel, the **Selection List** field shows the name of the selection list you are adding (the name you typed in the **List Name** field on the ADD SELECTION LIST panel).

If an ***** appears to the right of a subsystem name, the subsystem name no longer exists. To reduce confusion, blank out the **S** or **L**; when you do, a **D** replaces the **S** or **L** to indicate that the name has been deleted from the selection list.

1. In the **Selection Mode** field, enter **I** or **X**.

I Include the selected subsystems in the selection list
X Exclude the selected subsystems from the selection list (all subsystems not specifically excluded will be included)

2. Type **S, L,** or **M** (however, **M** is only valid if a device list exists) beside any of the subsystems listed on the panel (or type **S** beside the **Non-SVA Subsystems** field); then press **Enter**.

   S Select the subsystem to add to the selection list.

   Result: The SELECT SUBSYSTEMS panel remains displayed (Figure 7-10) so that you can make your next selection.

   L Create a selection list for that subsystem.

   Result: The SPECIFY DEVICES panel appears, as shown in Figure 7-11.

   M Modify a selection list.

   Result: The SPECIFY DEVICES panel appears, as shown in Figure 7-11.

```
                        *** SELECT DEVICES ***                ROW 1 OF 4
COMMAND===> _____________________________________________ SCROLL ===> PAGE

 Subsystem: IBMSYS08
 Selection List: XSUBSS1
 Selection Mode ===> _            I = include, X = exclude, D = delete
 Type S to select device

                 DEVICE                  DEVICE                  DEVICE
   DEV#  VOLSER  TYPE      DEV#  VOLSER  TYPE      DEV#  VOLSER  TYPE
   ----  ------  ------    ----  ------  ------    ----  ------  ------
 _ 011A  DATA01  33903   _ 0125  DATA05  33903   _ 0102  VOL112  SCSIA
 _ 011B  DATA02  33903   _ 0126  DATA06  33903   _ 0103  VOL113  SCSIA
 _ 011C  DATA03  33903   _ 0127  DATA07  33903   _ 0104  VOL114  SCSIA
 _ 011D  DATA04  33903   _ 0100  VOL110  SCSIA   _ 0105  VOL115  SCSIA
```

Figure 7-11. *Select Devices panel (SIBDL40)*

On this panel, the **Subsystem** and **Selection List** fields show the names you selected earlier. Use this list whenever you want to see what devices are attached to a subsystem.

3. Type either **I, X,** or **D** in the **Selection Mode** field and then press **Enter**.

   I Include the devices selected below in the selection list
   X Exclude the devices selected below from the selection list
   D Delete the device selection list for that subsystem

4. Type **S** beside any of the device addresses listed on the panel to select that device for inclusion or exclusion.

5. Issue the **END** command to return to the SELECT SUBSYSTEMS panel (Figure 7-10 on page 7-14), or issue the **RETURN** command to return to the SVAA main menu.

   Result: Reporter modifies or deletes the selection list; you return to the ADD SELECTION LIST panel (Figure 7-9 on page 7-13).

If you typed **S** beside **Specify Device Ranges** or **Specify Volser Ranges** on the ADD SELECTION LIST panel, Reporter handles each request in turn.

If you typed **L** beside one or more list names on the SELECT SUBSYSTEMS - LIST MAINTENANCE panel, Reporter handles each request in turn. Go to "Specifying Device Ranges" or "Specifying Volser Ranges" on page 7-17.

Specifying Device Ranges

The following panel appears when you type **S** beside **Specify Device Ranges** on the MODIFY SELECTION LIST panel.

```
                        *** SPECIFY DEVICE RANGES ***
 COMMAND===> _____________________________________________ SCROLL ===> PAGE

  Selection List: XSUBSS1

  Specify device ranges or FROM only for single device
  To include range, type I under ACTION
  To exclude range, type X under ACTION

             FROM   TO                FROM   TO                FROM   TO
      ACTION DEV#  DEV#       ACTION DEV#  DEV#       ACTION DEV#  DEV#
      ------ ----  ----       ------ ----  ----       ------ ----  ----

  ___
*****************************BOTTOM OF DATA*********************************
```

Figure 7-12. *Specify Device Ranges panel (SIBDL50)*

On this panel, the **Selection List** field shows the name of the selection list you are creating. When you first see such a panel, only one of the entry lines appears; it is blank. Use this panel when you know the ranges of unit addresses of the devices you are interested in.

1. On the current entry line, type the device ranges you want, and type **I** in the **ACTION** column to include the range or X to exclude the range. Use the line commands shown above to simplify entering the ranges.

   Notes:

   a. In typing device ranges, the **TO DEV#** field is optional. If you omit this field, the **FROM DEV#** field specifies a single unit address.

   b. Do not enter duplicate or conflicting entries.

   c. Do not specify both inclusion and exclusion of the same device.

   d. You can only use wildcard characters if you specify only the **FROM DEV#** field.

2. In the field at the left of a line, enter any of the insert, delete, or replicate line commands (see "Using Line Commands" on page 7-4 for details).

3. Press **Enter** cause the line commands to take effect.

4. Issue the **END** command to return to the previous panel, or issue the **RETURN** command to return to the SVAA main menu.

   Result: Your selection list contains the device ranges you entered.

Specifying Volser Ranges

The following panel appears when you type **S** beside **Specify Volser Ranges** on the ADD SELECTION LIST panel.

```
                         *** SPECIFY VOLSER RANGES ***
COMMAND===> ________________________________________________ SCROLL ===> PAGE

 Selection List: XSUBSS1

 Specify volser ranges or FROM only for single volser
 To include range, type I under ACTION
 To exclude range, type X under ACTION

             FROM     TO             FROM     TO              FROM     TO
     ACTION VOLSER  VOLSER   ACTION VOLSER  VOLSER    ACTION VOLSER  VOLSER
     ------ ------  ------   ------ ------  ------    ------ ------  ------

___     _   ______  ______      _   ______  ______       _   ______  ______
******************************BOTTOM OF DATA********************************
```

Figure 7-13. *Specify Volser Ranges panel (SIBDL80)*

On this panel, the **Selection List** field shows the name of the selection list you are creating. When you first see such a panel, only one of the entry lines appears. Use this panel when you know the volume serial numbers of the devices you are interested in.

Note: When you enter a volser range in the **FROM VOLSER** and **TO VOLSER** fields, the right-most series of digits define the range and the remaining digits to the left must be the same in each field (example: **AB1001** to **AB1007**.)

1. On the current entry line, type the volser ranges you want, and type **I** in the **ACTION** field to include the range or **X** to exclude the range. Use the line commands to simplify entering the ranges.

   Notes:

   a. In typing volser ranges, the **TO VOLSER** field is optional. If you omit this field, the **FROM VOLSER** field specifies a single volume.

   b. Do not enter duplicate or conflicting entries.

   c. Do not specify both inclusion and exclusion of the same device.

   d. You can only use wildcard characters if you use only the **FROM VOLSER** field.

2. In the field at the left of each line, enter any of the insert, delete, or replicate line commands (see “Using Line Commands” on page 7-4 for details).

3. Press **Enter** after using as many of these line commands as you need.

4. Issue the **END** command to return to the panel appropriate for your next action, or issue the **RETURN** command to return to the SVAA main menu.

   Result: Your selection list contains the volser ranges you entered.

Copying a Selection List

Sometimes, you need to create a new selection list that is similar to one you already have defined. The easiest way to do this is by copying the existing list to one with a new name, then modifying the new list.

To copy a selection list:

1. Type **C** on the MAINTAIN SELECTION LISTS panel (Figure 7-8 on page 7-13) to the left of the list you want to copy; then press **Enter**.

   Result: The COPY SELECTION LIST panel appears:

```
                        *** COPY SELECTION LIST ***
COMMAND===> ____________________________________________________________

  Source List Name: XSUBSS1
  Target List Name  ===>________
  Description  ===>_____________________________________________________
```

Figure 7-14. *Copy Selection List panel (SIBDL70)*

   On this panel, the **Source List Name** field contains the name of the list you selected for copying.

2. Type in the **Target List Name** field the name of the new selection list to which you want to copy the source list. This name is required.

   Note: List names cannot begin with **SIB**, which is reserved for internal use by SVAA.

3. Type up to 49 characters as a description of the new list. This description will appear to the right of the list name on future selection-list maintenance panels. This description is optional.

4. Press **Enter** to make the copy.

   Result: Reporter verifies the data you have entered.

5. Repeat steps 2, 3, and 4 to create any additional lists you need.

6. Issue the **END** command.

   Result: After a successful copy, Reporter displays the MODIFY SELECTION LIST panel so that you can make the changes needed to make the copied lists unique. Go to step 4 on page 7-19 to modify the lists.

Modifying a Selection List

To modify a selection list:

1. Select option **2** on the SVAA main menu to display the MAINTAIN SELECTION LISTS panel.
2. Type **m** on that panel beside the name of any selection list you want to modify.

   Result: The panel shown in Figure 7-15 on page 7-19 appears:

```
                          *** MODIFY SELECTION LIST ***
COMMAND===> _____________________________________________________________

 Selection List: XSUBSS1
 Description ===> Collect data from subsystem 1

 Type S to select one or more lists
 Type D to delete one or more lists

    _ Select Subsystems
    _ Specify Device Ranges
    _ Specify Volser Ranges
```

Figure 7-15. *Modify Selection List panel (SIBDL60)*

3. Optionally, enter a new description for the selection list.
4. Type **S** beside any of the types of selection lists you want to modify (subsystems, device ranges, or volser ranges).
5. Press **Enter**.

   Result: One of the following panels appears, depending on your choice:

SELECT SUBSYSTEMS panel	Go to “Selecting Subsystems and Devices” on page 7-20
SPECIFY DEVICE RANGES panel	Go to “Specifying Device Ranges” on page 7-22
SPECIFY VOLSER RANGES panel	Go to “Specifying Volser Ranges” on page 7-23

Selecting Subsystems and Devices

The following panel appears when you type **S** beside **Select Subsystems** on the MODIFY SELECTION LIST panel (Figure 7-15 on page 7-19).

```
            *** SELECT SUBSYSTEMS - LIST MAINTENANCE ***
 COMMAND===> _____________________________________________ SCROLL ===> PAGE


  Selection List: XSUBSS1
  Selection Mode ===>  _           I = include, X = exclude

  Type S to select subsystem
  Type L to create device selection list   (SVA subsystems only)
  Type M to modify device selection list   (SVA subsystems only)

    _  Non-SVA Subsystems

  If no subsystems are specifically marked, by default all existing
  SVA subsystems will be included for collection/reporting.

     _  ICESYS00       _  ICESYS01       _  ICESYS02       _  ICESYS03
     _  ICESYS04       _  ICESYS05       _  ICESYS06       _  ICESYS07
     _  ICESYS08       S  ICESYS09 *     _  ICESYS10       _  SVASYS15
     _  SVASYS12       S  SVASYS13 *     _  SVASYS14       _  SVASYS17
     _  SVASYS16       _  SVASYS17       _  SVASYS18       _  SVASYS21
     _  SVASYS20       _  SVASYS21       L  SVASYS22 *     _  SVASYS23
     _  SVASYS24       _  SVASYS25       _  SVASYS26       _  SVASYS29
*****************************BOTTOM OF DATA*********************************
```

Figure 7-16. *Select Subsystems - List Maintenance panel (SIBDL30)*

The **Selection Mode** field contains **I** or **X**, depending on how the list was created. The description field displays the description of that selection list.

To modify the selection list:

1. To delete the list of devices attached to a subsystem, blank the **L** to the left of the subsystem name.
2. In the **Selection Mode** field, type **I** or **X**:

   I Include the selected subsystems in the selection list
   X Exclude the selected subsystems from the selection list (all subsystems not specifically excluded will be included)

   Note: An * appearing to the right of an entry indicates that the selected subsystem is no longer defined. Blank the entry before continuing with the next step. A **D** replaces the * to indicate that the entry has been deleted.

3. Type **S, L,** or **M** (however, **M** is only valid if a selection list exists) beside any of the subsystems listed on the panel (or type **S** for non-STK systems).

   S Select the subsystem to add to the selection list.

   Result: The MODIFY SELECTION LIST panel remains displayed (Figure 7-15) so that you can make your next selection.

   L Create a device selection list.

   Result: The SELECT DEVICES panel appears, as shown in Figure 7-17 on page 7-21.

M Modify a device selection list.

Result: The SELECT DEVICES panel appears, as shown in Figure 7-17 on page 7-21.

```
                        *** SELECT DEVICES ***                ROW 1 OF 4
COMMAND===> _____________________________________________ SCROLL ===> PAGE

 Subsystem: IBMSYS08
 Selection List: XSUBSS1
 Selection Mode ===> _            I = include, X = exclude, D = delete
 Type S to select device

                 DEVICE                  DEVICE                  DEVICE
   DEV#  VOLSER  TYPE      DEV#  VOLSER  TYPE      DEV#  VOLSER  TYPE
   ----  ------  ------    ----  ------  ------    ----  ------  ------
 _ 011A  DATA01  33903   _ 0125  DATA05  33903   _ 0102  VOL112  SCSIA
 _ 011B  DATA02  33903   _ 0126  DATA06  33903   _ 0103  VOL113  SCSIA
 _ 011C  DATA03  33903   _ 0127  DATA07  33903   _ 0104  VOL114  SCSIA
 _ 011D  DATA04  33903   _ 0100  VOL110  SCSIA   _ 0105  VOL115  SCSIA
```

Figure 7-17. *Select Devices panel (SIBDL40)*

On this panel, the **Subsystem** and **Selection List** fields show the names you selected earlier. Use this list whenever you want to see what devices are attached to a subsystem.

4. Type either **I, X,** or **D** in the **Selection Mode** field and press **Enter**.

   I Include the devices selected below in the selection list
   X Exclude the devices selected below from the selection list
   D Delete the selection list for that subsystem

5. Type **S** beside any of the device addresses listed on the panel to select that device for inclusion or exclusion.

6. Issue the **END** command to return to the SELECT SUBSYSTEMS panel (Figure 7-10 on page 7-14), or issue the **RETURN** command to return to the SVAA main menu.

   Result: Reporter modifies or deletes the selection list; you return to the ADD SELECTION LIST panel (Figure 7-9 on page 7-13).

   If you typed **S** beside **Specify Device Ranges** or **Specify Volser Ranges** on the ADD SELECTION LIST panel, Reporter handles each request in turn.

   If you typed **L** beside one or more list names on the SELECT SUBSYSTEMS - LIST MAINTENANCE panel, Reporter handles each request in turn. Go to “Specifying Device Ranges” on page 7-22 or “Specifying Volser Ranges” on page 7-23.

Specifying Device Ranges

The following panel appears when you type **S** beside the **Specify Device Ranges** field on the MODIFY SELECTION LIST panel.

```
                        *** SPECIFY DEVICE RANGES ***
 COMMAND===> _____________________________________________ SCROLL ===> PAGE

  Selection List: XSUBSS1

  Specify device ranges or FROM only for single device
  To include range, type I under ACTION
  To exclude range, type X under ACTION

             FROM   TO                 FROM   TO                 FROM   TO
      ACTION DEV#  DEV#        ACTION DEV#  DEV#        ACTION DEV#  DEV#
      ------ ----  ----        ------ ----  ----        ------ ----  ----

  ___
*****************************BOTTOM OF DATA*********************************
```

Figure 7-18. *Specify Device Ranges panel (SIBDL50)*

On this panel, the **Selection List** field shows the name of the selection list you are modifying. When you first see such a panel, only one of the entry lines appears, unless some device ranges have already been specified, as when you are copying a list for subsequent modification.

1. If you are modifying a copied list, enter any of the line commands in the field at the left of a line (see “Using Line Commands” on page 7-4 for more information).
2. Press **Enter** when you have specified the desired device ranges.
3. Use the **Tab** key to get to the field in which you want to enter or modify a device range, type the device range you want, and type **I** in the **ACTION** field to include the range or **X** to exclude the range. Use line commands to simplify entering the ranges. In typing device ranges, the **TO DEV#** field is optional. (If you omit this field, the **FROM DEV#** field specifies a single unit address.)
4. Issue the **END** command to return to the previous panel, or issue the **RETURN** command to return to the SVAA main menu.

   Result: Your selection list contains the modified device ranges.

Specifying Volser Ranges

The following panel appears when you type **S** beside the **Specify Volser Ranges** field on the MODIFY SELECTION LIST panel.

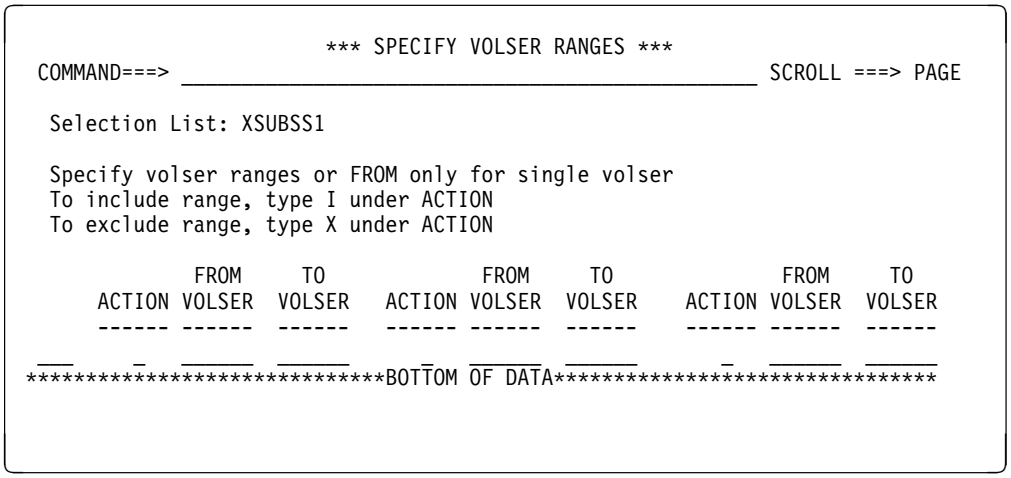


Figure 7-19. *Specify Volser Ranges panel (SIBDL80)*

On this panel, the **Selection List** field shows the name of the selection list you are modifying. When you first see this panel, only one of the entry lines appears, unless some volser ranges have already been specified, as when you are copying a list for subsequent modification.

1. If you are modifying a copied list, enter any of the line commands in the field at the left of a line (for more information, see “Using Line Commands” on page 7-4).
2. Press **Enter** when you have specified the desired volser ranges.
3. Use the **Tab** key get to the field in which you want to enter or modify a volser range, type the volser range you want, and type **I** in the **ACTION** field to include the range or X to exclude the range. Use the line commands shown above to simplify entering the ranges.
4. Issue the **END** command to return to the previous panel, or issue the **RETURN** command to return to the SVAA main menu.

   Result: Your selection list contains the modified volser ranges.

Deleting a Selection List

To delete a selection list and all devices associated with the list:

1. Select option **2** on the SVAA main menu to display the MAINTAIN SELECTION LISTS panel.

```
                    *** MAINTAIN SELECTION LISTS ***
 COMMAND===> _____________________________________________ SCROLL ===> PAGE

  To add a selection list, use the ADD command or PF key

   List Commands:  B - (Browse)     C - (Copy)
                   D - (Delete)     M - (Modify)

   LIST NAME       DESCRIPTION
   ---------       -----------
   _ XSUBSS1       Collect data from subsystem 1
   _ XSUBSS2       Collect data from subsystem 2
   _ IDEVSS3       Include certain devices for subsystem 3
*****************************BOTTOM OF DATA*********************************
```

Figure 7-20. *Maintain Selection Lists panel (SIBDL10)*

   On this panel, the **LIST NAME** column is a list of names of selection lists. To the left of each name is a field in which you type one of the list commands shown in the panel.

2. Type **D** beside the names of the selection lists you want to delete.

   Result: Reporter deletes the list and displays a message to that effect near the upper-left corner of the panel.

3. Issue the **END** command to return to the previous panel, or issue the **RETURN** command to return to the SVAA main menu.

   CAUTION:
   There is no way to retrieve a selection list once it has been deleted.

Browsing a Selection List

To browse a selection list:

1. Select option **2** on the SVAA main menu to display the MAINTAIN SELECTION LISTS panel, as shown in Figure 7-20.

2. Type **B** on the MAINTAIN SELECTION LISTS panel (Figure 7-8 on page 7-13) to the left of the list you want to browse; then press **Enter**.

   Result: THE SELECT LISTS - LIST MAINTENANCE panel appears, as shown in Figure 7-21 on page 7-25.

```
                    *** SELECT LISTS - LIST MAINTENANCE ***
COMMAND===> ____________________________________________________________
 BROWSE

 Selection List: XSUBSS1

 Type S to select one or more options

   _ Select Subsystems
   _ Specify Device Ranges
   _ Specify Volser Ranges
```

Figure 7-21. *Select Lists - List Maintenance panel (SIBRX50)*

Note that **BROWSE** appears in the upper-left corner. You cannot enter data on panels with **BROWSE** present, but you can enter letters to select options.

3. Type **S** beside one or more of the device list types you want to browse (subsystem, device range, or volser range). Press **Enter**.

   Result: One of the following panels appears, depending on your choice:

SELECT SUBSYSTEMS panel	Go to “Selecting Subsystems and Devices”
SPECIFY DEVICE RANGES panel	Go to “Specifying Device Ranges” on page 7-27
SPECIFY VOLSER RANGES panel	Go to “Specifying Volser Ranges” on page 7-27

Selecting Subsystems and Devices

The following panel appears when you type **S** beside the **Select Subsystem** field on the SELECT LISTS - LIST MAINTENANCE panel.

```
            *** SELECT SUBSYSTEMS - LIST MAINTENANCE ***
COMMAND===> ____________________________________________ SCROLL ===> PAGE
 BROWSE

 Selection List: XSUBSS1
 Selection Mode: X              I = include, X = exclude

    Type B over L to browse device selection list (SVA subsystems only)

      Non-SVA Subsystems

    If no subsystems are specifically marked, by default all existing
    SVA subsystems will be included for collection/reporting.

    S  SVASYS00          SVASYS01          SVASYS02          SVASYS03
       SVASYS04          SVASYS05          SVASYS06          SVASYS07
    L  SVASYS08          SVASYS09          SVASYS10          SVASYS15
       SVASYS12          SVASYS13        S SVASYS14          SVASYS17
       SVASYS16          SVASYS17          SVASYS18          SVASYS21
       SVASYS20          SVASYS21          SVASYS22          SVASYS25
       SVASYS24        S SVASYS25          SVASYS26          SVASYS29
```

Figure 7-22. *Select Subsystems - List Maintenance panel (SIBRX00).* This panel shows the names of all subsystems named in the selection list.

An L beside a subsystem name indicates that you have created a selection list for that subsystem. If an * appears to the right of a name beside which there is an L, the subsystem name cannot be found. An S indicates that the subsystem is included in or excluded from the selection list named at the top of the panel.

1. Type **B** over the L beside any of the existing selection lists you wish to browse.
2. Issue the **END** command.

   Result: The SELECT DEVICES - LIST MAINTENANCE panel appears, as shown in Figure 7-23.

```
                *** SELECT DEVICES - LIST MAINTENANCE ***
COMMAND===> _____________________________________________ SCROLL ===> PAGE
BROWSE

 Subsystem: SVASYS08
 Selection List: XSUBSS1
 Selection Mode: X                I = include, X = exclude

                 DEVICE                   DEVICE                   DEVICE
   DEV#  VOLSER  TYPE      DEV#  VOLSER  TYPE      DEV#  VOLSER  TYPE
   ----  ------  ------    ----  ------  ------    ----  ------  ------
 S 011A  DATA01  33903     0125  DATA15  33903     0104  VOLMAY  SCSIA
   011B  DATA02  33903     0126  DATA16  33903     0105  VOLJUN  SCSIA
   011C  DATA03  33903     0127  DATA17  33903     0106  VOLJUL  SCSIA
 S 011D  DATA04  33903     0100  VOLJAN  SCSIA     0107  VOLAUG  SCSIA
   011E  DATA05  33903     0101  VOLFEB  SCSIA     0108  VOLSEP  SCSIA
   011F  DATA06  33903   S 0102  VOLMAR  SCSIA     0109  VOLOCT  SCSIA
   012A  DATA11  33903     0103  VOLAPR  SCSIA     0110  VOLNOV  SCSIA
```

Figure 7-23. *Select Devices - List Maintenance panel (SIBRX10)*

This panel shows you the list of functional devices in all the selection lists you selected. An S indicates that the functional device is included in or excluded from the selection list named at the top of the panel. An * appearing to the right of a list indicates that the selected functional device no longer exists in the subsystem. You should delete the entry when you next modify the list.

3. Issue the **END** command to return to the panel appropriate for your next action, or issue the **RETURN** command to return to the SVAA main menu.

Specifying Device Ranges

The following panel appears when you type **S** beside **Specify Device Ranges** on the SELECT LISTS - LIST MAINTENANCE panel (Figure 7-21).

```
                *** SPECIFY DEVICE RANGES - LIST MAINTENANCE ***
COMMAND===> _____________________________________________ SCROLL ===> PAGE
 BROWSE

 Selection List: XSUBSS1

              I = include range      X = exclude range

            FROM   TO               FROM   TO               FROM   TO
      ACTION DEV#  DEV#       ACTION DEV#  DEV#       ACTION DEV#  DEV#
      ------ ----  ----       ------ ----  ----       ------ ----  ----
         I   011A  011F          I   0127  012A          X   0112  0118
```

Figure 7-24. *Specify Device Ranges - List Maintenance panel (SIBRX20)*

This panel displays the ranges of unit addresses selected for the subsystem.

1. For each range, **I** indicates the range is included; **X** indicates it is excluded.
2. Issue the **END** command to return to the panel appropriate for your next action, or issue the **RETURN** command to return to the SVAA main menu.

Specifying Volser Ranges

The following panel appears when you type **S** beside **Specify Volser Ranges** on the SELECT LISTS - LIST MAINTENANCE panel (Figure 7-21).

```
          *** SPECIFY VOLSER RANGES - LIST MAINTENANCE ***
COMMAND===> _____________________________________________ SCROLL ===> PAGE
 BROWSE

 Selection List: XSUBSS1

                I = include range      X = exclude range

            FROM     TO             FROM     TO             FROM     TO
     ACTION VOLSER   VOLSER   ACTION VOLSER   VOLSER   ACTION VOLSER   VOLSER
     ------ ------   ------   ------ ------   ------   ------ ------   ------
        I   DIV100   DIV199      X   DIV175   DIV179      I   DIV300   DIV399
        I   DIV450   DIV599      I   DIV750   ______      I   DIV755   ______
        I   DIV800   DIV999      X   DIV890   DIV899      X   DIV920   DIV924
```

Figure 7-25. *Specify Volser Ranges - List Maintenance panel (SIBRX60)*

This panel displays the ranges of volume serial numbers you have selected for this subsystem.

1. For each range, **I** indicates the range is included; **X** indicates it is excluded.
2. Issue the **END** command to return to the panel appropriate for your next action, or issue the **RETURN** command to return to the SVAA main menu.

Data Collection and Reporting

To select Subsystem Reporting:

1. Select option **3** from the SVAA main menu.

   Result: The SVAA SUBSYSTEM REPORTING panel appears, as shown in Figure 7-26.

```
SIBDL00                *** SVAA SUBSYSTEM REPORTING ***
OPTION ===>

   1  PT DATA       - Collect Performance Tracking (PT) data
   2  OR DATA       - Collect On Request (OR) data
   3  PERFORMANCE   - Report subsystem performance
   4  UTILIZATION   - Report subsystem space utilization
   5  EXTRACTION    - Create SAS extraction program
```

Figure 7-26. *SVAA Subsystem Reporting menu (SIBDL00)*

2. On this panel, type a number in the OPTION field:

   1 Collect Performance Tracking (PT) data (see below)
   2 Collect On Request (OR) data (see page 7-33)
   3 Report performance (see page 7-37)
   4 Report space utilization (see page 7-48)
   5 Extract history data (see page 7-50)

3. Press **Enter**.

   Result: See the indicated page in this chapter.

4. Issue the **END** command to return to the SVAA main menu.

Collecting Performance Tracking (PT) Data

This section explains how to use the SVAA panels to control Performance Tracking (PT) data collection. Read “Collecting Subsystem Data” on page 2-2 before using this section. (Chapter 8, “SVAA Subcommands and Commands” explains how to use subcommands and batch programs for the same purpose.)

This section describes how to specify:

- The devices for which Reporter is to collect data
- The time periods for collection
- The intervals for collection, and how collection is to be synchronized
- Whether device performance or cache effectiveness data (or both) are to be collected

Note: Collected data is stored in the logging files, as specified during SIBSRP initialization with the INIT MAIN subcommand.

Figure 7-27 on page 7-29 shows the panels you use to control PT data collection.

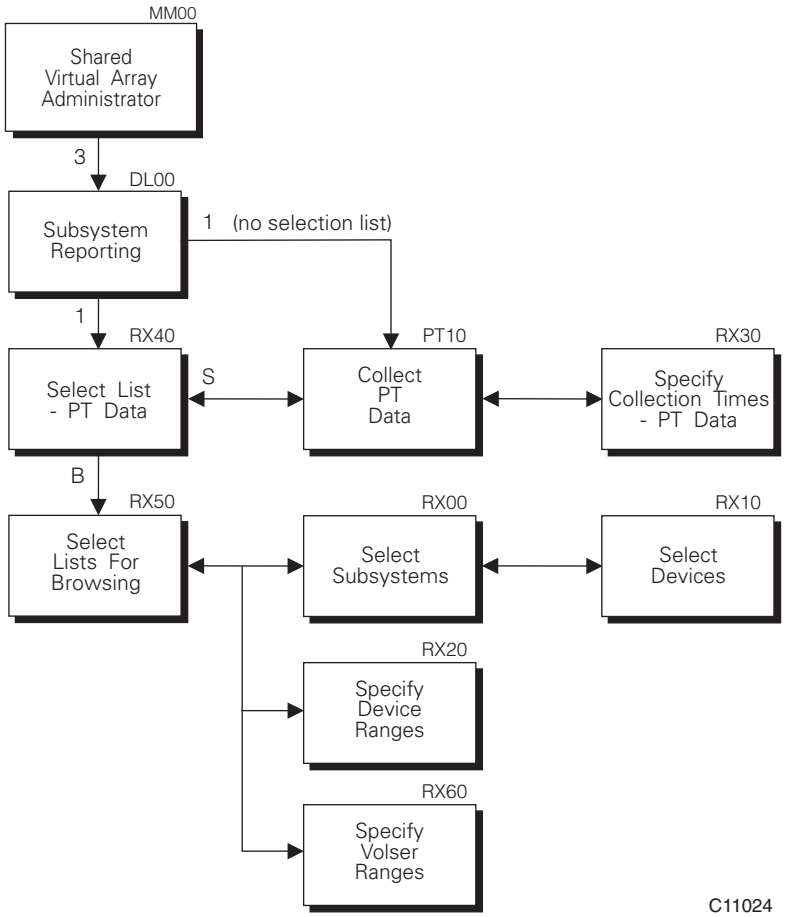


Figure 7-27. *Panels used in specifying PT data collection*

To specify PT data collection:

1. Select **1** from the SVAA SUBSYSTEM REPORTING panel (see Figure 7-26 on page 7-28).

   Result: If there are no selection lists defined, Reporter displays the COLLECT PT DATA panel (Figure 7-29 on page 7-30). If any selection lists exist, Reporter displays the SELECT LIST - COLLECT PT DATA panel, as shown in Figure 7-28 on page 7-30.

```
                *** SELECT LIST - PT DATA  ***
COMMAND===> _____________________________________________ SCROLL ===> PAGE

  Press Enter to bypass list selection or type the
  following list commands and then press Enter.

  List Commands:  B - (Browse)     S - (Select)

  LIST NAME       DESCRIPTION
  ---------       -----------
  _ XSUBSS1       Subsystem1 data will not be collected
  _ XSUBSS2       Subsystem2 data will not be collected
  _ IDEVSS3       Include certain devices for subsystem 3
```

Figure 7-28. *Select List - PT Data panel (SIBRX40)*

2. If you do not want to select a device selection list, then press **Enter**.

   Result: Reporter is to collect data for all SVA subsystems. The COLLECT PT DATA panel appears (go to step 6 on page 7-31).

3. Enter either **B** or **S** beside one or more of the list names.

   B Displays the names of the functional devices in the list. See “Browsing a Selection List” on page 7-24 for more information.

   Note: You can only browse (view) selection lists while in this mode.

   S Selects the subsystem for which data is to be collected.

4. Press **Enter**.

   Result: If you typed **B**, Figure 7-28 reappears when you have finished browsing.

   If you typed **S**, the COLLECT PT DATA panel appears:

```
                      *** COLLECT PT DATA ***
COMMAND===> ____________________________________________________________

 Selection List: XSUBSS1

    Times      ===>  N        (Type O for time options)

    Collection Interval  (select one of the following)
       RMF ===> __       Minutes ===> 0020       Hours ===> ____

    Synchronization Time  ===> NO minutes

    Device Performance    ===> Y
    Cache Effectiveness   ===> Y
    Modify                ===> N
```

Figure 7-29. *Collect PT Data panel (SIBPT10)*

This panel appears when a selection list was chosen on the SELECT LIST- PT DATA panel, or when there are no selection lists. This panel appears with default values for the fields, as shown in Figure 7-29.

5. To specify up to 16 pairs of starting and ending times for collection, change the N in the **Times** field to O. When you press **Enter**, Reporter displays the SPECIFY COLLECTION TIMES - PT DATA panel for you to enter the time ranges. If you leave this field containing **N**, collection starts immediately and continues indefinitely. On return from the SPECIFY COLLECTION TIMES - PT DATA panel, Reporter sets this field to **Y** (provided you specify valid data there).

6. Change any of the default values to select the data you want. The following table shows the values you can specify for each field.

Collection Interval RMF	Enter the 2-digit suffix of the RMF startup member. (See "Synchronization of Collection Interval with RMF" on page 4-10.)
Collection Interval Minutes	Enter a number **1** to **1440**
Collection Interval Hours	Enter a number **1** to **24**
Synchronization Time	Enter a number **0** to **59**, or **NO**
Device Performance	Enter **Y** or **N**
Cache Effectiveness	Enter **Y** or **N**

7. Change the **Modify** field to **Y** to change an existing collection run. (See "Starting Data Collection" on page 4-2 for more information.)

8. Press **Enter**.

   Result: If you Typed **O** in the **Times** field, SPECIFY COLLECTION TIMES panel appears, as shown in Figure 7-34 on page 7-36.

```
              *** SPECIFY COLLECTION TIMES - PT DATA ***
COMMAND===> ____________________________________________________________

Specify one or more starting-time/ending-time pairs for collection

    Start Time  To  End Time                Start Time  To  End Time
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
```

Figure 7-30. *Specify Collection Times - PT Data panel (SIBRX30)*

9. Enter the starting and ending times of up to 16 time ranges that you want collection to cover (see page 6-6 for information about time ranges). Use two digits for the hour and two digits for the minutes after the hour that collection is to start. Use leading zeroes, if necessary (for example, 0330). Do not overlap time ranges. Starting times must be between **0000** and **2359**; ending times must be between **0001** and **2400**.

10. Issue the **END** command.

    Result: The COLLECT PT DATA panel appears again, as in Figure 7-29 on page 7-30.

11. On that panel, issue the **SUBMIT** command to start the collection run.
12. If you entered **S** beside other list names on the SELECT LIST - PT DATA panel, Reporter displays the COLLECT PT DATA panel again for you to specify collection for each of those lists.
13. Otherwise, issue the **END** command again to return to the SELECT LIST panel, and issue the **END** command once more to return to the SVAA SUBSYSTEM REPORTING panel.

Collecting On Request (OR) Data

This section explains how to use the SVAA panels to control On Request (OR) data collection. Read “Collecting Subsystem Data” on page 2-2 before using this section. (Chapter 8, “SVAA Subcommands and Commands” explains how to use subcommands or a batch program to control data collection).

Figure 7-31 shows the panels you use to specify the data to be collected.

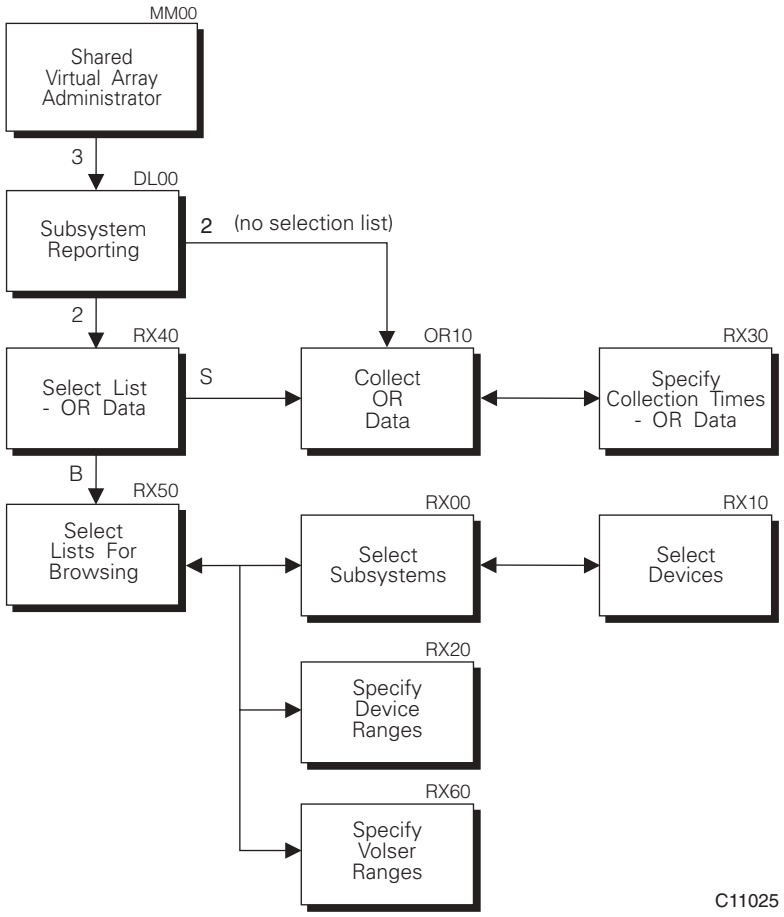


Figure 7-31. *Panels used in specifying OR data collection*

To specify OR data collection:

1. Select **2** from the SVAA SUBSYSTEM REPORTING panel (see Figure 7-26 on page 7-28).

   Result: If there are no selection lists defined, Reporter displays the COLLECT OR DATA panel (Figure 7-33 on page 7-35).

   If any selection lists exist, Reporter displays the SELECT LIST - OR DATA panel, as shown in Figure 7-32 on page 7-34.

```
                *** SELECT LIST - OR DATA ***
COMMAND===> _____________________________________________ SCROLL ===> PAGE

  Press Enter to bypass list selection or type the
  following list commands and then press Enter.

  List Commands   B - (Browse)     S - (Select)

  LIST NAME       DESCRIPTION
  ---------       -----------
  _ XSUBSS1       Subsystem1 data will not be collected
  _ XSUBSS2       Subsystem2 data will not be collected
  _ IDEVSS3       Include certain devices for subsystem 3
```

Figure 7-32. *Select List - OR Data panel (SIBRX40)*

2. If you do not want to use a device selection list, press **Enter**.

   Result: Reporter collects data for all the SVA functional devices and subsystems available to it. The COLLECT OR DATA panel appears with the message **"NO LIST SELECTED"** in the **Selection List** field (go to step 5 on page 7-35).

3. Type either **B** or **S** beside one or more of the list names.

   B Displays the names of the devices in the list. See "Browsing a Selection List" on page 7-24 for more information.

   Note: You can only browse (view) selection lists while in this mode.

   S Selects the subsystem for which data is to be collected.

4. When all values are entered, press **Enter**.

   Result: If you typed **B**, the panel shown in Figure 7-32 reappears when you have finished browsing.

   If you typed **S**, the COLLECT OR DATA panel appears in turn for each list beside which you entered **S**.

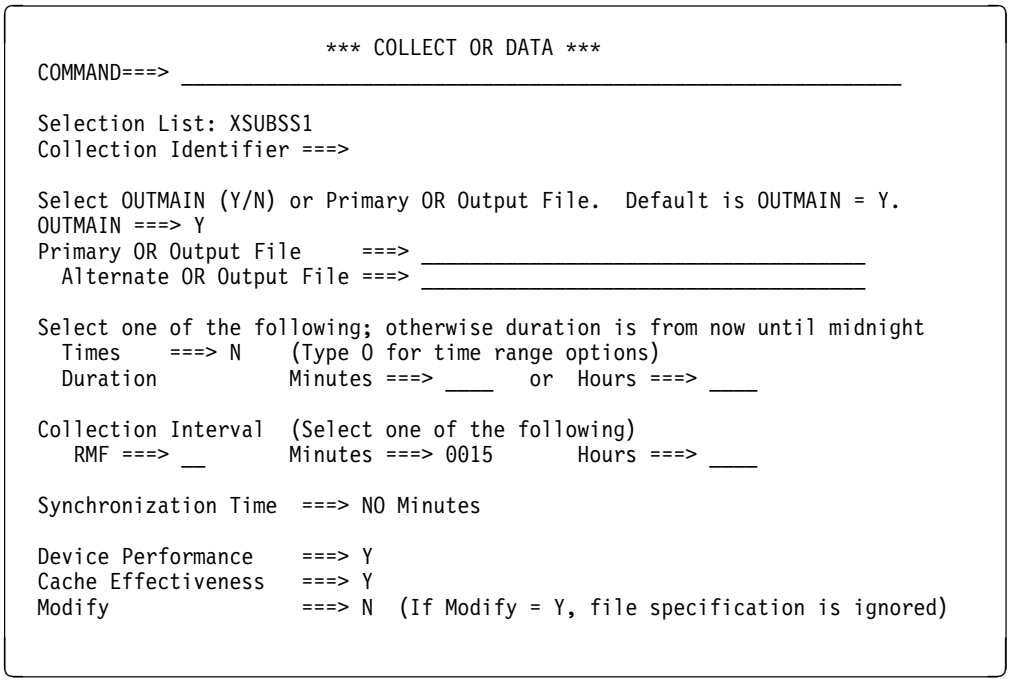


Figure 7-33. *Collect OR Data panel (SIBOR10).* This panel appears with default values for the fields.

This panel appears when a selection list was chosen on the SELECT LIST - OR DATA panel, or when there are no selection lists. The selection list you chose appears in the **Selection List** field. If you did not specify a list on the previous panel, this field contains "No list selected" when the panel appears and Reporter collects data for all devices.

5. Specify the collection identifier, a name for this on-request collection run. Use up to 16 characters, a-z, A-Z, 0-9, and $, @, #, -, +, _, &, ., or /; do not use **PERFORMANCETRACK**.

   Note: If you are specifying a new collection run, (by **MODIFY=N** on this panel), Reporter stores your collection identifier, with the case set according to the current setting of the SET CASE subcommand, converting the identifier to uppercase if appropriate.

   If you are modifying an existing collection run (by **MODIFY=Y** on this panel), use the exact uppercase and lowercase characters that you entered when you defined the collection run you are modifying. In these circumstances, Reporter ignores the setting of the SET CASE subcommand.

6. Specify the output files for this collection run, either with a **Y** in the **OUTMAIN** field to specify the main logging file, or with a file name in one of the **OR Output File** fields. If you leave both fields blank, Reporter uses the main logging file.

7. Specify a value in either the **Times** or **Duration** field. Change the **Times** field to **0** to request a panel on which you can specify up to 16 time ranges for collection. Or, for a duration of collection, specify up to 9999 in either the **Minutes** or **Hours** field.

   If you leave both fields blank, the collection duration is from the time of this request until midnight.

8. Change any of the default values in the remaining fields to select the report you want. The following table shows the values you can specify for each field and shows where you can get more information.

Collection Interval RMF	Enter the 2-digit suffix of the RMF startup member. (See "Synchronization of Collection Interval with RMF" on page 4-10.)
Collection Interval Minutes	Enter **1** to **1440**.
Collection Interval Hours	Enter **1** to **24**.
Synchronization Time	Enter **0** to **59**, or **NO**.
Device Performance	Enter **Y** or **N**.
Cache Effectiveness	Enter **Y** or **N**.

9. Change the **Modify** field to **Y** to change an existing collection run. If you specify MODIFY Y, Reporter ignores all file specifications on this panel. (See "Starting Data Collection" on page 4-2 for more information.)
10. If you do not want to specify time ranges for data collection, go to step 13.
11. Press **Enter**.

    Result: The SPECIFY COLLECTION TIMES - OR DATA panel appears, as shown in Figure 7-34.

```
                *** SPECIFY COLLECTION TIMES - OR DATA ****
COMMAND===> ____________________________________________________________

Specify one or more starting-time/ending-time pairs for collection

    Start Time  To  End Time                Start Time  To  End Time
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
```

Figure 7-34. *Specify Collection Times - OR Data panel (SIBRX30)*

12. Enter the starting and ending times of up to 16 time ranges that you want collection to cover (see page 6-6 for information about time ranges). Use two digits for the hour and two digits for the minutes after the hour that collection is to start. Use leading zeroes, if necessary (for example, 0330). Do not overlap time ranges. Starting times must be between **0000** and **2359**; ending times must be between **0001** and **2400**.
13. Issue the **END** command.

    Result: The COLLECT OR DATA panel appears again, as in Figure 7-33 on page 7-35.

14. On that panel, issue the **SUBMIT** command to start the collection run.

15. If you entered **S** beside other list names on the SELECT LIST - OR DATA panel, Reporter displays the COLLECT OR DATA panel again for you to specify collection for each of those lists.
16. Otherwise, issue the **END** command again to return to the SELECT LIST - OR DATA panel, and issue the **END** command once more to return to the SVAA SUBSYSTEM REPORTING panel.

Specifying Performance Reporting

Figure 7-35 shows the panels you use to specify performance reporting:

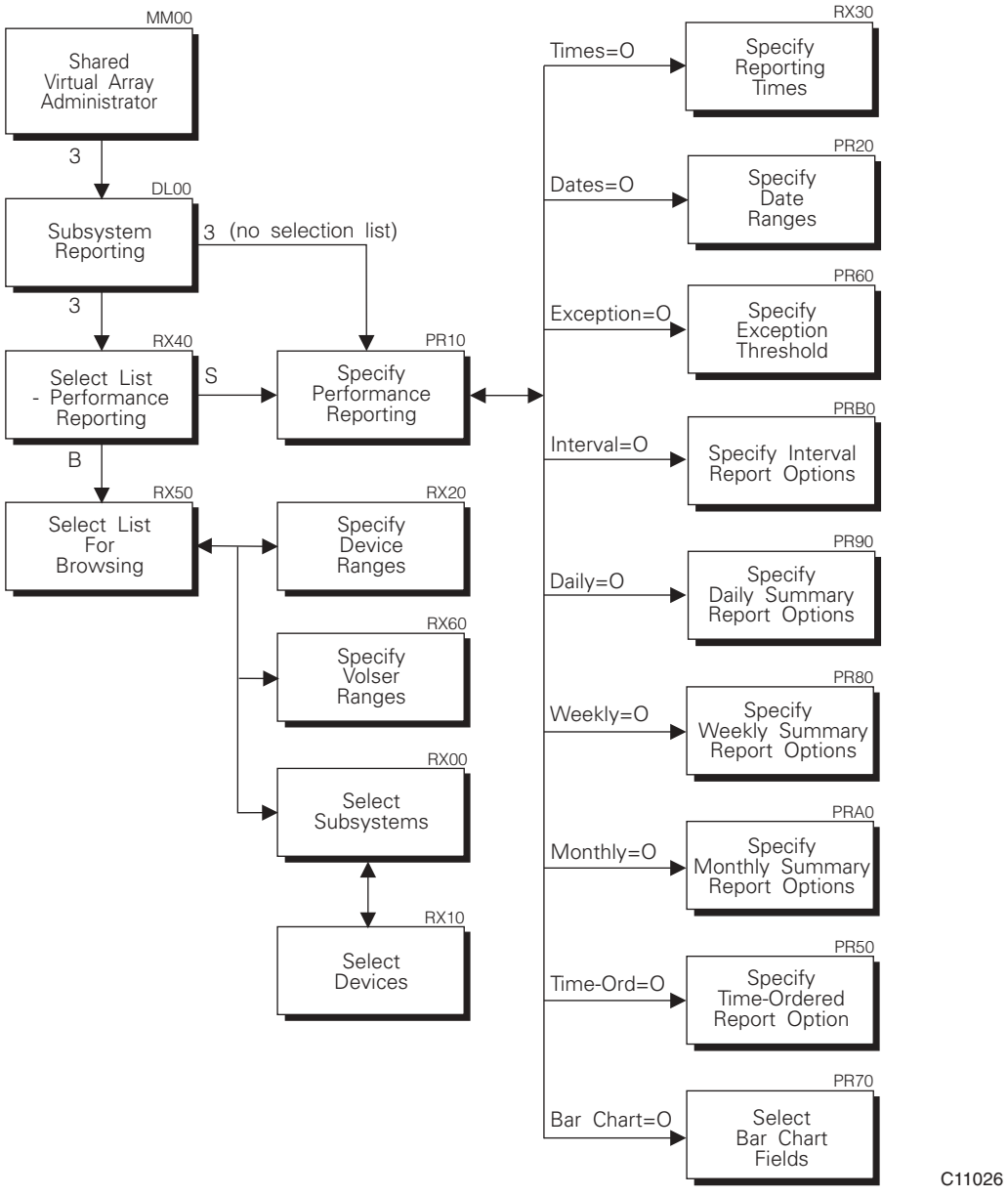


Figure 7-35. *Panels used in specifying performance reporting*

To request performance reporting:

1. Choose option **3** on the SVAA SUBSYSTEM REPORTING panel.

   Result: If no selection lists are defined, Reporter displays the SPECIFY PERFORMANCE REPORTING panel (Figure 7-37 on page 7-39).

   If any selection lists exist, Reporter displays the SELECT LIST- PERFORMANCE REPORTING panel, as shown in Figure 7-36:

```
                *** SELECT LIST - PERFORMANCE REPORTING ***
COMMAND===> _____________________________________________ SCROLL ===> PAGE

  Press Enter to bypass List Selection or select the
  following list commands and then press enter.

  List Commands:  B - (Browse)     S - (Select)

  LIST NAME       DESCRIPTION
  _ XSUBSS1       Subsystem1 data will not be reported
  _ XSUBSS2       Subsystem2 data will not be reported
  _ IDEVSS3       Include certain devices for subsystem 3
```

Figure 7-36. *Select List - Performance Reporting panel (SIBRX40)*

2. Optionally, type **B** to browse the selection lists. You can browse the lists before selecting them, to make sure that they list the subsystems and functional devices that you want to include in the report.

   Result: Reporter displays the SELECT LISTS - LIST MAINTENANCE panel (Figure 7-21 on page 7-25). Go to the sequence of steps beginning after that figure.

3. Either:

   - Press **Enter** without selecting a list name to specify that reporting is for all subsystems.

     or

   - Type **S** beside the selection list for which you want performance reporting; then press **Enter**.

   Result: Reporter displays the SPECIFY PERFORMANCE REPORTING panel, as shown in Figure 7-37 on page 7-39.

```
                *** SPECIFY PERFORMANCE REPORTING ***
COMMAND===> ____________________________________________________________
Selection List: XSUBSS1

INFILE DSN or SAS LIB DSN ===> ____________________________________________
SAS File Name             ===> _______

Collection Identifier     ===> PERFORMANCETRACK

Report Interval  Minutes  ===>  120      or   Hours ===> ____

Device Details     ===> N   SVAA SMF Number ===> ___   Suppress Print ===> N

Device Performance ===> Y   Cache Effectiveness ===> Y

Type O for options:
 Times ===> N       Dates ===> N

Report Types       (Type Y to select report, O for report options)
 Exception ===> N   Overall ===> Y   Interval     ===> N   Daily     ===> N
 Weekly    ===> N   Monthly ===> N   Time-Ordered ===> N   Bar Chart ===> N
```

Figure 7-37. *Specify Performance Reporting panel (SIBPR10).* If you are displaying 24 PF4 keys, the last two lines of this panel are not visible. Use a PFSHOW OFF command to access these fields.

This panel appears with the selection list you chose appearing in the first field and with default values as shown in the various fields.

4. Enter up to 44 characters for either:

   - A data set name for the input file that contains the raw collected data on which the report is to be based (enter the name in the **INFILE DSN** or **SAS LIB DSN** field). If you use the name you entered in a SAS/C library, you must enter the SAS file name (up to seven characters).

     Result: Reporter automatically creates a name for the SAS file containing the summarized history data (use the **INFILE DSN** or **SAS LIB DSN** field for the library name and the **SAS File Name** field for the member name).

     Note: You cannot use wildcard characters in the filespecs.

   - Reporter automatically creates a SAS program to process the data, as well as the JCL to run the program.

     Go to step 5.

5. If the report is based on on-request data collection, change the ID in the **Collection Identifier** field to the appropriate collection-run identifier.

6. Specify a value, either **1** to **1440** in the **Report Interval - Minutes** or **1** to **24** in the **Report Interval - Hours** field.

   Default: 120 minutes.

7. To obtain a detailed report for the devices in the list, change the Device Details value to Y.

   Default: No device details (overall report only).

8. Optionally, enter the SVAA SMF number 128 to 255. (This is normally not required.)

Default: None

9. To prevent the report from being printed, change the **Suppress Print** value to **Y**. You can use the Reporter output later to create SAS/GRAPH displays.

   Note: You must also specify a name in the **SAS Device Detail File** and the **SAS Subsystem Totals File** fields (or both) on the options panel appropriate for the report. This saves the report output for SAS; it is used as input to the graphic reports.

10. If you do not want device performance reports, change the **Device Performance** field to **N**.

11. If you do not want cache effectiveness reports, change the **Cache Effectiveness** field to **N**.

    Note: For each field, in steps 12 through 14, Reporter displays the panel on which you can specify your choices for that option. Issue the **END** command after specifying the options on each panel. Reporter then displays the next panel for you to specify options, and, after the last, displays the SPECIFY PERFORMANCE REPORTING panel.

12. To enter time ranges, change the **Times** field value to the letter **0**. Then press **Enter** to display the SPECIFY REPORTING TIMES - PERFORMANCE REPORTING panel (Figure 7-38 on page 7-41). Go to step 1 below that panel.

    Default: Reporter reports on all the data in the file.

13. To specify dates for which data is to be reported, change the **Dates** field to **0**. Then press **Enter** to display the SPECIFY DATES panel (Figure 7-39 on page 7-42). Go to "Specifying Date Ranges" on page 7-42.

    Default: Report all data.

14. Select the report types for which you want to specify options by changing the **N** value to **0** near the bottom of the SPECIFY PERFORMANCE REPORTING panel (except for the **Overall** field). If you want a particular report but do not want to modify its options, change the corresponding **N** value to **Y**.

    Result: Reporter displays in turn panels on which you specify the options:

Options for	Go to
Exception	"Specifying Exception Thresholds" on page 7-43
Overall	N/A
Interval	"Specifying Interval Report Options" on page 7-43
Daily	"Specifying Summary Report Options" on page 7-45
Weekly	"Specifying Summary Report Options" on page 7-45
Monthly	"Specifying Summary Report Options" on page 7-45
Time-ordered	"Specifying Time-Ordered Report Options" on page 7-46
Bar chart	"Specifying Bar-Chart Values" on page 7-47

    Note: If you select any of these reports except Bar Chart, you must also select either Device Performance or Cache Effectiveness.

15. After specifying the options on each panel, issue the **END** command.

    Result: Reporter displays the next panel for which you have requested to specify options.

    Note: If you specify **0** in the **Overall** field, Reporter places the cursor under that field and displays the error message, **Valid selection options are Y or N**.

16. After the last of the options panels, Reporter displays the SPECIFY PERFORMANCE REPORTING panel again. Notice that Reporter has changed the **0** characters that you used to indicate that you wanted to specify options to **Y** (Reporter assumes that because you specified options for a particular report, then you want that report). If you did not specify options, then Reporter changes the character back to **N**.

    At this time, you can change any of the fields to **0** again to verify that your options are as you want them or to change the options to other values.

17. When you are satisfied that you have correctly specified all the options, select the report types you want by the changing the appropriate **N** values to **Y** near the bottom of the SPECIFY PERFORMANCE REPORTING panel, and issue the **SUBMIT** command.

    Result: The message "Request report" appears in the upper-right corner of the panel to indicate that Reporter has accepted your request. Reporter creates a batch job to run the SAS program.

18. Issue the **END** command to return to the main panel.

    Result: The SVAA main panel reappears and you can either choose another option or exit from Reporter panels.

Specifying Reporting Times

The following panel appears when you type **0** in the **Times** field on the SPECIFY PERFORMANCE REPORTING panel.

```
              *** SPECIFY REPORTING TIMES - PERFORMANCE REPORTING ***
COMMAND===> _____________________________________________________________

Specify one or more starting-time/ending-time pairs for reporting

    Start Time  To  End Time                Start Time  To  End Time
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
      __:__         __:__                    __:__         __:__
```

Figure 7-38. *Specify Reporting Times panel (SIBRX30)*

1. Enter the starting and ending times of up to 16 time ranges that you want reporting to cover (see “Specifying Time Ranges for Collection” on page 4-5 for more information). Use two digits for the hour and two digits for the minutes after the hour that collection is to start. Use leading zeroes, if necessary (for example, 0330). Do not overlap time ranges. Starting times must be between **0000** and **2359**; ending times must be between **0001** and **2400**.

2. Issue the **END** command.

    Result: The SPECIFY PERFORMANCE REPORTING panel appears again, as in Figure 7-37 on page 7-39.

3. Go to step 15 on page 7-40.

Specifying Date Ranges

The panel in Figure 7-39 appears when you type **0** in the **Dates** field on the SPECIFY PERFORMANCE REPORTING panel.

```
                          *** SPECIFY DATE RANGES ***
COMMAND===>

 Selection Mode: I                I = include, X = exclude

 Type "S" next to selected days.

 ===>   MON     TUE     WED     THU     FRI     SAT     SUN


 Enter one or more start-stop date pairs for inclusion or exclusion.
 (From and to dates are specified in the format DDMMMYYYY).

    From-Date          To-Date            From-Date          To-Date
  Day Mon Year      Day Mon Year        Day Mon Year      Day Mon Year
    01JAN1999         01JUN1999
    _________         _________          _________         _________
    _________         _________          _________         _________
    _________         _________          _________         _________
    _________         _________          _________         _________
    _________         _________          _________         _________
    _________         _________          _________         _________
    _________         _________          _________         _________
    _________         _________          _________         _________
```

Figure 7-39. *Specify Date Ranges panel (SIBPR20)*

Use this panel either to include or to exclude reporting dates for all reports. For some reports, you can override these general dates with dates for those reports only.

If you specify exclusion, the reports will cover all days and dates in the collected data except those days or dates you specify here.

1. Type **X** or **I** in the **Selection Mode** field to exclude or include the date ranges that you will enter in step 3. (For general information, see “Choosing Report Dates or Date Ranges” on page 6-8.)
2. Type **S** next to the days of the week that you want to include in or exclude from reporting. (You can either exclude or include days on any one panel—you cannot do both.)
3. Enter up to 16 date ranges for which you want reports. Starting and ending dates are inclusive.

   Specify dates in the form *ddmmmyyyy* (for example, **04DEC1998**). Alternatively, use **-n** to indicate a number of days previous to the current date; use **0** to indicate the current date. Future days are allowed.

   Default: All data in the input file.
4. Issue the **END** command.

   Result: The SPECIFY PERFORMANCE REPORTING panel appears again, as in Figure 7-37 on page 7-39.

5. Go to step 15 on page 7-40.

Specifying Exception Thresholds

The following panel appears when you type **0** in the **Exception** field on the SPECIFY PERFORMANCE REPORTING panel.

```
                    *** SPECIFY EXCEPTION THRESHOLDS ***
COMMAND===> _____________________________________________________________

 Enter partition (all, production, test) ===> PRODUCTION
 Exception File Name ===> _____________________________________      (or)
 Enter threshold value or "*" for default threshold.

                          THRESHOLD                                THRESHOLD
VALUE TO COMPARE          LT     GT     VALUE TO COMPARE           LT     GT
 I/O Rate (device)        ____   ____    I/O rate (subsystem)      ____   ____
 Device Available %       ____   ____    I/O Hit %                 ____   ____
 Drive Mod Coeff Var      ____   ____    Read Hit %                ____   ____
 Access Density           ____   ____    Write Hit %               ____   ____
 I/O Service Time         ____   ____    DASD Fast Write           ____   ____
 Disconnect Time          ____   ____       Constraint (device)    ____   ____
 Device Utilization %     ____   ____    DASD Fast Write
 Disconnect Time %        ____   ____       Constraint (subsystem) ____   ____
 Channel I/O Rate         ____   ____    State Rate (device)
 % Active on Channel      ____   ____    Stage Rate (subsystem)    ____   ____
 Drive Module Util *      ____   ____    Tracks Occupied           ____   ____
 Net Capacity Load %      ____   ____    Hits per Stage            ____   ____
                                            (xx.x or xxxx)
```

Figure 7-40. *Specify Exception Thresholds panel (SIBPR60)*

On this panel you specify exception thresholds. (For general information, see “Specifying Exception Thresholds” on page 6-10.)

1. Specify the partitions for which you want reporting in the **Enter Partition** field.
2. Specify the name of any exception thresholds file (for information, see “Specifying Exception Thresholds in a File” on page 6-11).
3. Enter the exception thresholds you wish to have reported; the values you specify here will override those in the thresholds file for the same variable. Enter an * to accept the default value for a threshold. (For default threshold values, see Appendix B, “Exception Thresholds.”)
4. Issue the **END** command.

   Result: The SPECIFY PERFORMANCE REPORTING panel appears again, as in Figure 7-37 on page 7-39.
5. Go to step 15 on page 7-40.

Specifying Interval Report Options

The following panel appears when you type **0** in the **Interval** field on the SPECIFY PERFORMANCE REPORTING panel.

```
          *** SPECIFY INTERVAL REPORT OPTIONS ***
COMMAND===> ____________________________________________________________


   SVAA Graphics Files:

      SAS Device Details File
         SAS Library DSN  ===> ____________________________________________
         SAS File Name    ===> _________

      Generations to Keep ===> 5


      SAS Subsystem Totals File
         SAS Library DSN  ===> ____________________________________________
         SAS File Name    ===> _________

      Generations to Keep ===> 5
```

Figure 7-41. *Specify Interval Report Options panel (SIBPRB0)*

1. Specify the names of the files that are to contain input to SAS/GRAPH reports. You can specify files for device details, or subsystem totals, or both. If you omit this information, Reporter does not save data for graphics display.

   a. Enter up to 44 characters in the **SAS Library DSN** field for the name of the SAS data set that is to contain the file.

   b. Enter up to 8 characters in the **SAS File Name** field for the name of the file that is to contain the SAS input.

2. For each file, specify the number of generations of SAS data to keep (0 to 999). This enables you to keep several versions of the output.

3. Issue the **END** command.

   Result: The SPECIFY PERFORMANCE REPORTING panel appears again, as in Figure 7-37 on page 7-39.

4. Go to step 15 on page 7-40.

Specifying Summary Report Options

If you typed **0** in the **Daily, Weekly,** or **Monthly** field on the SPECIFY PERFORMANCE REPORTING panel, the SPECIFY DAILY SUMMARY REPORT OPTIONS panel appears, or the corresponding WEEKLY or MONTHLY report options panel, as shown in Figure 7-42.

```
              *** SPECIFY DAILY SUMMARY REPORT OPTIONS ***
COMMAND===> ____________________________________________________________
   Report overrides:
      Include dates (enter one or more start/end date pairs for
                     inclusion in report):

      From-Date      To-Date              From-Date      To-Date
    Day Mon Year  Day Mon Year          Day Mon Year  Day Mon Year
      _________    _________            __________    __________
      _________    _________            __________    __________

  SVAA Graphics Files:

     SAS Device Details File:
         SAS Library DSN ===> ____________________________________________
         SAS File Name   ===> _______
     Generations to Keep ===> 5

     SAS Subsystem Totals File:
         SAS Library DSN ===> ____________________________________________
         SAS File Name   ===> _______
     Generations to Keep ===> 5
```

Figure 7-42. *Specify Daily Summary Report Options panel (SIBPR90).* With the exception of the title lines, the panels for specifying weekly and monthly summaries are identical to this panel.

1. Specify the ranges of dates that you want the report to include. You can specify only four date ranges on this panel. For this report only, these dates take the place of the dates on the SPECIFY DATE RANGES panel.

   Default: Reporting is for all dates for which data exists.

2. Specify the names of the files that are to contain input to SAS/GRAPH reports. You can specify files for device details or for subsystem totals, or both.

   a. Enter up to 44 characters in the **SAS Library DSN** field for the name of the SAS data set that is to contain the file.

   b. Enter up to 8 characters in the **SAS File Name** field for the name of the file that is to contain the SAS input.

3. For each file, specify the number of generations of SAS data to keep. You can specify a number from 0 to 999 inclusive.

4. Issue the **END** command.

   Result: The SPECIFY PERFORMANCE REPORTING panel appears again, as in Figure 7-37 on page 7-39.

5. Go to step 15 on page 7-40.

Specifying Time-Ordered Report Options

The following panel appears when you type **0** in the **Time-Ordered** field on the SPECIFY PERFORMANCE REPORTING panel.

```
                *** SPECIFY TIME-ORDERED REPORT OPTIONS ***
COMMAND===> ____________________________________________________________

   SVAA Graphics Files:

      SAS Device Details File:
         SAS Library DSN  ===> ____________________________________
         SAS File Name    ===> _______

      Generations to Keep ===> 2



      SAS Subsystem Totals File:
         SAS Library DSN  ===> ____________________________________
         SAS File Name    ===> _______

      Generations to Keep ===> 2
```

Figure 7-43. *Specify Time-Ordered Report Options panel (SIBPR50)*

1. Specify the names of the files that are to contain input to SAS/GRAPH reports. You can specify files for device details, subsystem totals, or both.

   a. Enter up to 44 characters in the **SAS Library DSN** field for the name of the SAS data set that is to contain the file.

   b. Enter up to 8 characters in the **SAS File Name** field for the name of the file that is to contain the SAS input.

2. For each file, specify the number of generations of SAS data to keep.

3. Issue the **END** command.

   Result: The SPECIFY PERFORMANCE REPORTING panel appears again, as in Figure 7-37 on page 7-39.

4. Go to step 15 on page 7-40.

Specifying Bar-Chart Values

The following panel appears when you type **0** in the **Bar Chart** field on the SPECIFY PERFORMANCE REPORTING panel.

```
             *** SELECT BAR CHART FIELDS ***
COMMAND===> _____________________________________________________________

Type S to select bar chart fields:

_  Device Available %              _  Drive Module Utilization %
_  I/O Rate                        _  Drive Module Coeff of Var
_  Access Density                  _  I/O Hit %
_  Kilobytes per Second            _  Read Hit %
_  I/O Service Time                _  Write Hit %
_  Disconnect Time                 _  DASD Fast Write Constraint
_  Device Utilization %            _  Stage Rate
_  Disconnect Time %               _  Hits per Stage
_  Free Space Collection Load      _  Tracks Occupied
_  Net Capacity Load %
```

Figure 7-44. *Select Bar Chart Fields panel (SIBPR70).* The fields on this panel are explained in Appendix C, “Variables for Bar Chart Reports.”

1. Choose reporting values for which you want bar-chart reports by typing an **S** beside the name of the value. For more information about bar charts, see Appendix C, “Variables for Bar Chart Reports.”
2. Issue the **END** command.

   Result: The SPECIFY PERFORMANCE REPORTING panel appears again, as in Figure 7-37 on page 7-39.
3. Go to step 15 on page 7-40.

Specifying Space Utilization Reporting

Figure 7-45 shows the panels you use to specify the data to be collected in a report:

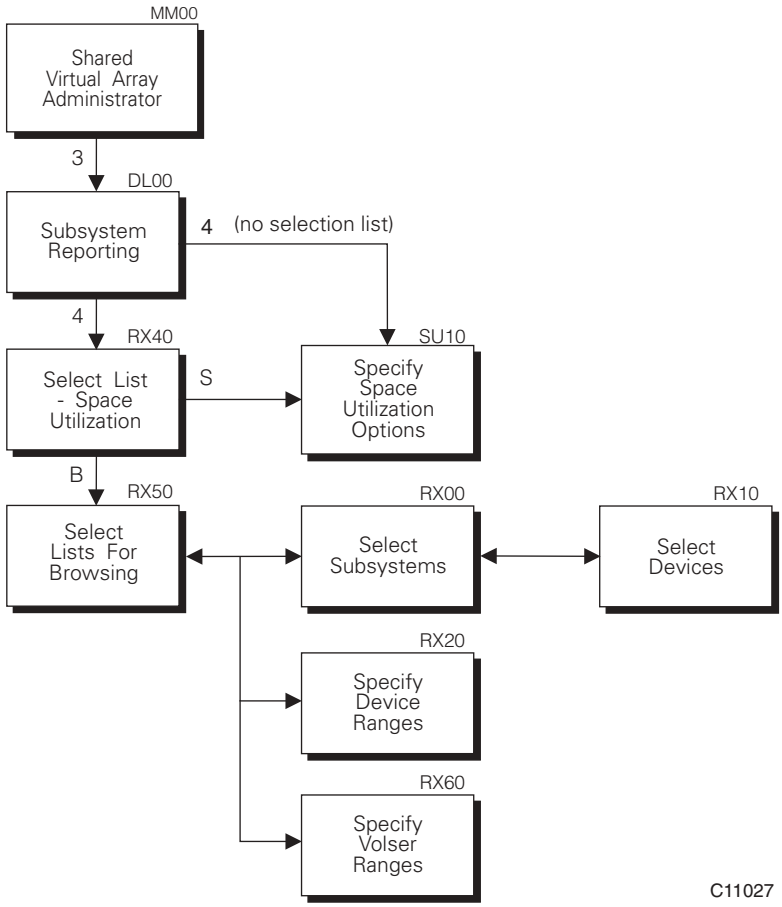


Figure 7-45. *Panels used in specifying space utilization reporting*

To request Space Utilization reporting:

1. Choose option **4** on the SVAA SUBSYSTEM REPORTING panel (Figure 7-26 on page 7-28).

   Result: If there are no selection lists defined, Reporter displays the SELECTION LIST - SPACE UTILIZATION REPORTING panel, as shown in Figure 7-47 on page 7-49 (instead of a name in the **Selection List** field, that panel displays **"NO LIST WAS SELECTED"** in the field.

   If any selection lists exist, Reporter displays the SELECT LIST - PERFORMANCE REPORTING panel, as shown in Figure 7-36 on page 7-38.

```
          *** SELECT LIST - SPACE UTILIZATION ***
COMMAND===> _____________________________________________ SCROLL ===> PAGE

  Press enter to bypass device list selection or type the
  following list commands and then press Enter.

  List Commands:  B - (Browse)     S - (Select)

  LIST NAME       DESCRIPTION
  _ XSUBSS1       Subsystem1 data will not be collected
  _ XSUBSS2       Subsystem2 data will not be collected
  _ IDEVSS3       Include certain devices for subsystem 3
```

Figure 7-46. *Select List - Space Utilization panel (SIBRX40)*

2. Optionally, you can browse a list before selecting it, to make sure that the list contains the subsystems and functional devices that you want to include in the report.

   Result: Reporter displays the SELECT LISTS - LIST MAINTENANCE panel (Figure 7-21 on page 7-25). Go to the sequence of steps beginning after that figure.

3. Either:

   - Press **Enter** without selecting a list name to specify that reporting is for all SVA subsystems,
     or
   - Type **S** beside the selection lists for which you want Space Utilization reporting. Then press **Enter**.

   Result: Reporter displays the SPACE UTILIZATION REPORTING panel, as shown in Figure 7-47.

```
              *** SPECIFY SPACE UTILIZATION REPORTING ***
COMMAND===> ____________________________________________________________

 Selection List: XSUBSS1

   To submit, use the SUBMIT command or PF key
```

Figure 7-47. *Specify Space Utilization Reporting panel (SIBSU10)*

   Issue the **SUBMIT** command to request the report.

4. When the “submitted” message appears in the upper-right corner, issue the **END** command to return to select another device for Space Utilization reporting, or issue the **RETURN** command to return to the SVAA SUBSYSTEM REPORTING panel.

Extracting Summarized History Data

This section explains how to use the SVAA panels to control data extraction from the summarized history file. Read “Extracting History Data for Graphing” on page 5-7 before using this section. You cannot use subcommands or a batch program to tailor data extraction.

Figure 7-48 on page 7-51 shows the panels you use to specify data extraction.

Note: For non-SVA subsystems, only cache-effectiveness data is available in the summarize history file. Statistics not available for these subsystems appear as zeroes in the extracted data file.

To specify data extraction:

1. Select **5** from the SVAA SUBSYSTEM REPORTING panel (see Figure 7-26 on page 7-28).

   Result: If no selection lists are defined, Reporter displays the CREATE SAS EXTRACTION PROGRAM panel (Figure 7-50). Go to step 5 on page 7-52.

   If any selection lists exist, Reporter displays the SELECT LIST - DATA EXTRACTION panel, as shown in Figure 7-49.

```
SIBRX40          *** SELECT LIST - DATA EXTRACTION ***            ROW 1 OF 8
COMMAND===>                                                 SCROLL ===> PAGE

  Press Enter to bypass list selection or type the
  following list commands and then press Enter.

  List Commands:   B - (Browse)      S - (Select)

   LIST NAME        DESCRIPTION
   ---------        -----------
     A
     B
     C
     D
****************************** BOTTOM OF DATA ********************************
```

Figure 7-49. *Select List - Data Extraction panel (SIBRX40)*

2. If you decide that you do not want to use a device selection list, press **Enter**.

   Result: Reporter extracts data for all the SVA functional devices and subsystems available to it. The CREATE SAS EXTRACTION PROGRAM panel appears with the message **“NO LIST SELECTED”** in the **Selection List** field (go to step 5 on page 7-52).

3. To browse or select selection lists, type either **B** or **S** beside one or more of the list names.

   B Displays the names of the devices in the list. See “Browsing a Selection List” on page 7-24 for more information.

   Note: You can only browse (view) selection lists while in this mode.

   S Selects the list of devices for which data is to be extracted.

4. When you have entered all the values, press **Enter**.

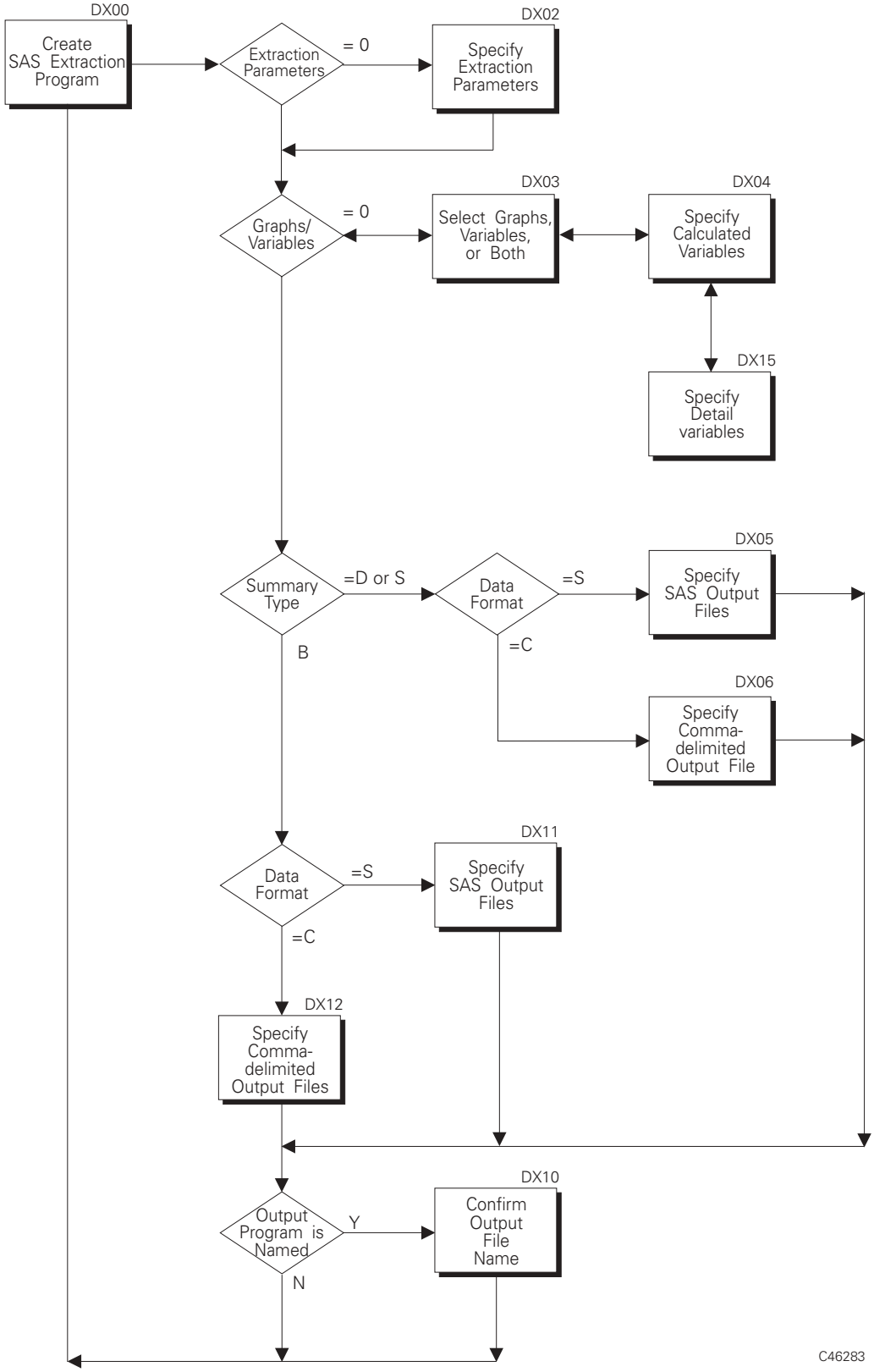


Figure 7-48. *Panels used in specifying data extraction*

Result: If you typed **B**, the panel shown in Figure 7-49 on page 7-50 reappears when you have finished browsing.

If you typed **S** to select a list, the CREATE SAS EXTRACTION PROGRAM panel appears. Figure 7-50 appears.

```
SIBDX00              *** CREATE SAS EXTRACTION PROGRAM ***
COMMAND===>

Selection List: No list selected

Input SAS history DSN   ===>
 SAS history file name  ===>

Output program library  ===>
 Member name            ===>

Extraction parameters   ===>     (O=Options/D=Display parameters)

Graphs and/or variables ===>     (O=Options/D=Display selections)

Summary type            ===> B   (D=Device/S=Subsystem/B=Both)

Extracted data format   ===>     (C=Comma-delimited/S=SAS)
             SAS format specified
```

Figure 7-50. *Create SAS Extraction Program panel (SIBDX00).* The message below the last field appears only when you return to this panel after executing all the secondary panels; it indicates your selection for the **Extracted data format** field.

The panel in Figure 7-50 appears when a selection list was chosen on the SELECT LIST - DATA EXTRACTION panel, or when there are no selection lists. The first selection list you chose, if any, appears in the **Selection List** field. If you did not specify a list on the previous panel, this field contains **`No list selected`** when the panel appears and Reporter extracts data for all devices.

Note: You can press **Enter** after entering your choices in any of the fields on this panel. Alternatively, you can fill in all the choices and then press **Enter**. Whichever procedure you choose, SVAA displays the appropriate panels for you to complete your specification of data extraction, and then returns you to the CREATE SAS EXTRACTION PROGRAM panel. (The following directions assume that you press **Enter** after filling in all the fields.)

5. Specify the source of the data to be extracted:

   - Enter the name of the SAS history data set and the SAS history file from which data is to be extracted.

     Default: None. This name is required.

6. Specify where the SAS program is to be stored:

   - Optionally enter the name of the library and the name of the member that is to contain the output program and JCL.

     Default: If you do not supply the names, SVAA does not save the SAS program and the JCL.

     Default: None.

7. In the **Extraction parameters** field, specify one of:

   O Specifies that you wish to define the options for this choice.

   D Displays the currently selected parameters.

   Result: SVAA displays the information at the bottom of the panel.

8. In the **Graphs and/or variables** field, specify one of:

   O Specifies that you wish to define the options for this choice.

   D Displays the currently selected types of graphs and variables.

   Result: SVAA displays the information at the bottom of the panel.

9. In the **Summary type** field, specify the type of summary you want by entering one of:

   D Extracts data for each functional device.

   S Extracts data for each subsystem. This is the default.

   B Extracts data for both functional devices and subsystems.

10. In the **Extracted data format** field, specify the format in which you want the extracted data to be stored:

    C SVAA stores the data in comma-delimited format for processing with a suitable program, such as Microsoft Excel, Borland Paradox, or Lotus 1-2-3.

    S SVAA stores the data in SAS format for processing with SAS or SAS/GRAPH.

11. Press **Enter**

    Result: SVAA displays secondary panels for the fields on which you have not already pressed **Enter**, as in the following steps.

12. When you select a summary:

    - By **either** device or subsystem (step 9) and
    - With output in **comma-delimited** files (step 10),

    SVAA enables you to specify the names of the output files. If you do not make a selection, SVAA writes the output to the same files that you specified the last time you used this panel. (If you did not do so, go to step 13 on page 7-54.) In OS/390, Figure 7-51 appears.

```
SIBDX06              *** SPECIFY COMMA-DELIMITED OUTPUT FILE ***
COMMAND===>

Extracted Data DSN ===>
```

Figure 7-51. *Specify Comma-Delimited Output File panel (subsystem or device) (SIBDX06)*

   a. On this panel, specify the name of the file to be used for the extracted data.

Defaults:

- None (you must supply a data set name)

b. Press **Enter**.

Result: SVAA proceeds to the next step below.

13. When you select a summary:

- By **either** device or subsystem (step 9 on page 7-53) and
- With output in **SAS** files (step 10 on page 7-53),

SVAA enables you to specify the names of the output files. (If you did not make such a selection, go to step 14.) Figure 7-52 appears

```
SIBDX05                 *** SPECIFY SAS OUTPUT FILES ***
COMMAND===>

Specify one or both:

 SAS extracted data DSN ===>
  File name             ===>

 SAS transport DSN      ===>
```

Figure 7-52. *Specify SAS Output Files panel (subsystem or device) (SIBDX05)*

Specify either or both of the SAS extracted data file and the SAS transport file (neither is required):

a. Specify the name of the SAS extracted data file.

Defaults:

- None (but if you do not specify a name, SVAA does not produce an extracted SAS file)

b. Specify the name of the SAS transport file.

Defaults:

- None.

c. Press **Enter**.

Result: SVAA proceeds to the next step below.

14. When you select a summary:

- By **both** device and subsystem (step 9 on page 7-53) and
- With output in **comma-delimited** files (step 10 on page 7-53),

SVAA displays a panel for you to specify the names of the output files. Figure 7-53 appears. (If you did not make such a selection, go to step 15 on page 7-55.)

```
SIBDX12              *** SPECIFY COMMA-DELIMITED OUTPUT FILES ***
COMMAND===>

Specify dataset names:

 Device detail    ===>

 Subsystem totals ===>
```

Figure 7-53. *Specify Comma-Delimited DSN (subsystem and detail) panel (SIBDX12)*

a. Specify the name you want for the device-detail data.

   Defaults:

   - None (you must supply a data set name)

b. Specify the name you want for the subsystem-totals data.

   Defaults:

   - None (you must supply a data set name)

c. Press **Enter**.

   Result: SVAA proceeds to the next step below.

15. When you select a summary:

    - By **both** device or subsystem (step 9 on page 7-53) and
    - With output in **SAS** files (step 10 on page 7-53),

    SVAA enables you to specify the names of the output files. Figure 7-54 appears. (If you did not make such a selection, go to step 16 on page 7-56.)

```
SIBDX11                *** SPECIFY SAS OUTPUT FILES ***
COMMAND===>

Specify dataset names:

 Device detail:
  SAS extracted data DSN ===>
   File name             ===>

  SAS transport DSN      ===>

 Subsystem totals:
  SAS extracted data DSN ===>
   File name             ===>

  SAS transport DSN      ===>
```

Figure 7-54. *Specify SAS Output Files panel (subsystem and detail) (SIBDX11)*

Specify either or both of the extracted data file and the SAS transport file for device details and subsystem totals:

a. On this panel, specify for both devices and subsystems the names of the output files.

   Defaults for extracted data:

   - None (but if you do not specify a name, SVAA does not produce an extracted SAS file)

   Defaults for the SAS transport file:

   - None.

b. Specify the names of the SAS transport files to be used.

c. Press **Enter**.

   Result: SVAA proceeds to the next step below.

16. If you specified **0** in the **Extraction parameters** field of the **Create SAS Extraction Program** panel, SVAA displays the following panel:

```
SIBDX02              *** SPECIFY EXTRACTION PARAMETERS ***
COMMAND===>

Start Date     ===> -7          Stop Date  ===> -1
Start Time     ===>    :        Stop Time  ===>    :

Summarization Interval   (To modify, enter one of the following)
 Minutes       ===> 120
 Hours         ===>

Partition      ===> P     (O=Overall/P=Production/T=Test)
```

Figure 7-55. *Specify Extraction Parameters panel (SIBDX02)*

a. Enter the ranges of dates for which you want data to be extracted.

   Default: -7:-1 (that is, the previous week, not including today)

b. Enter the ranges of times for which you want data to be extracted.

   Default: All times in the history file.

c. Enter the summarization interval in either the **Minutes** or the **Hours** field (the default is 120 minutes).

   Note: The interval you specify must be a multiple of the interval in the history file and must be a factor of 1440 (if you specify minutes) or of 24 (if you specify hours).

d. Specify the partitions for which data extraction is to apply by entering one of:

   0 Extract overall data for all partitions.

   P Extract data for the Production partition. This is the default.

   T Extract data for the Test partition.

e. Press **Enter**.

   Result: SVAA proceeds to the next step below.

17. If you specified **0** in the **Graphs and/or variables** field, the following panel appears:

```
SIBDX03             *** SELECT GRAPHS, VARIABLES OR BOTH ***
COMMAND===>

Select StorageTek-supplied graphs:     (Y=Yes/N=No)
  Avg service time and avg transfer size         ===> Y
  Avg service time and cache hit percent         ===> Y
  Avg service time and data throughput           ===> Y
  Avg service time and I/O rate                  ===> Y
  Avg service time for top 25 functional devices ===> Y
  I/O service time at max I/O rate               ===> Y
  SVA free space analysis                        ===> Y


Select custom variables:      (O=Options/X=Exclude custom selection)
  Channel             ===>
  Cache Effectiveness ===>
  Device Performance  ===>


Estimated workstation space requirement:
```

Figure 7-56. *Specify Graph Types, Variables or Both panel (SIBDX03)*

a. Enter **Y** beside any of the supplied graphs you wish to specify.

Table 7-2 shows the graphs you can select.

Note: If you select a type of graph, you do not need to specify the variables needed—SVAA does that automatically.

Table 7-2 (Page 1 of 2). *Supplied sample graphs*

Name of graph	Description	Variables used
Average service time and average transfer size	Shows, over time: • Average connect times per I/O • Average disconnect times per I/O • Average I/O transfer sizes	iostdisc, iostconn, kbperio, enddate, endtime, subsname, partmemb
Average service time and cache hit %	Shows, over time: • Average connect time per I/O • Average disconnect time per I/O • Cache hit percentages	iostdisc, iostconn, iohit, enddate, endtime, subsname, partmemb
Average service time and data throughput	Shows, over time: • Average connect time per I/O • Average disconnect time per I/O • Kilobytes per second	kbytesec, iostconn, iostdisc, enddate, endtime, subsname, partmemb
Average service time and I/O rate	Shows, over time: • Average connect time per I/O • Average disconnect time per I/O • Average I/O rate	iorate, iostconn, iostdisc, enddate, endtime, subsname, partmemb

Table 7-2 (Page 2 of 2). *Supplied sample graphs*

Name of graph	Description	Variables used
Average service time for top 25 functional devices (those that have the most I/O)	Shows, for the 25 busiest functional devices: • Average connect time per I/O • Average disconnect time per I/O • Percentage of I/Os for the volume to the total I/Os	subdurat, devutlti, conntime, devactiv, enddate, endtime, subsname, volser, partmemb
I/O service time at maximum I/O rate	Shows, over time: • Average connect time per I/O at maximum I/O rate • Average disconect time per I/O at maximum I/O rate • Maximum I/O rate for the period	maxiort, maxconn, maxdisc, enddate, endtime, subsname, partmemb
SVA free space analysis	Shows, over time: • Net capacity load percentage • Uncollected free space percentage • Total physical capacity	ncapldp, totbecp, puncfsp, enddate, endtime, subsname, partmemb

b. In the **Select custom variables** fields:

- Type **0** beside any type of variable to see a list of the variables of that type that you can include.
- Type **X** beside any type of variable you had previously selected but no longer want.

If you leave the entry field blank, SVAA assumes you want that type of variable if **Selected** appears to the right of the field (this indicates that you previously selected that type of variable).

c. Press **Enter**.

Result: If you specify any of the supplied graphs, SVAA creates programs to extract data for the graphs you specify.

18. If you specify any types of custom variables, SVAA displays the following panels, in order corresponding to the types of variables you specify in the **Select Custom Variables** insert. See Appendix E, “Using Data Extraction” for descriptions of all the variable names.

Note: These panels list all the calculated variables, as well as those of the detail variables that you are most likely to need, such as those that identify the data in some way. Only the first few variables are shown in the following panel examples.

```
SIBDX04              *** CHANNEL VARIABLES ***                      ROW 1 OF 20
 COMMAND===>                                                    SCROLL ===> PAGE

 Select from detail list ===>      Selected     (O=Options/X=Exclude detail)

 ================================================================================

 Commands:   R - (Remove)    S - (Select)

     Variable     Description                                          Message
     --------     -----------                                          --------
  _  actvchnl     % subsys active on channel
  _  chansped     Channel speed in MB per second
  _  ciorate      Channel I/O rate
 ******************************* BOTTOM OF DATA ********************************
```

Figure 7-57. *Channel Variables panel (SIBDX04)*

```
SIBDX04              *** CACHE-EFFECTIVENESS VARIABLES ***          ROW 1 OF 39
 COMMAND===>                                                    SCROLL ===> PAGE

 Select from detail list ===>      Selected     (O=Options/X=Exclude detail)

 ================================================================================

 Commands:   R - (Remove)    S - (Select)

     Variable     Description                                          Message
     --------     -----------                                          --------
  _  collid       Collection identifier
  _  cpuid        ID of CPU performing data collection
  _  devnum       Functional device number as known to host
⋮
```

Figure 7-58. *Cache-Effectiveness Variables panel (SIBDX04)*

```
SIBDX04              *** DEVICE-PERFORMANCE VARIABLES ***           ROW 1 OF 40
 COMMAND===>                                                    SCROLL ===> PAGE

 Select from detail list ===>      Selected     (O=Options/X=Exclude detail)

 ================================================================================

 Commands:   R - (Remove)    S - (Select)

     Variable     Description                                          Message
     --------     -----------                                          --------
  _  accden       I/O per second per GB functional
  _  capacity     Functional device capacity in GB
  _  coeffvr      % drive module coeff of variation (production)
⋮
```

Figure 7-59. *Device-Performance Variables panel (SIBDX04)*

a. As each of these panels appear, indicate that you want to specify variables by entering in the **Detail variables** field one of:

   0 Specify detail variables for the corresponding type of custom variables.

   Result: SVAA displays the corresponding detail variables panel (see step 19). When you return from the detail variables panel, **Selected** appears to the right of the **Detail variables** field, if any detail variables were selected.

   X Exclude detail variables from the output.

   Note: The variables you have selected on the detail variables panels remain selected even though they are to be excluded from the current extraction program.

b. Select the calculated variables or identification variables to be included in the extracted data by entering beside the variable names one of:

   S Include the calculated variable or identification variable in data extraction.

   Result: If you press **Enter** at this point, **Selected** appears to the right of each variable name you selected.

   R Remove an earlier selection of the calculated variable.

c. When you have finished specifying variables, press **Enter**.

   Result: SVAA displays the panels on which you can specify the detail variables you have requested.

19. If you specify **0** in the **Detail variables** field of any of the custom variables panels, SVAA displays the following panels in order corresponding to the custom variables panels. See Appendix E, “Using Data Extraction” for descriptions of all the variable names.

```
SIBDX15          *** DETAIL CHANNEL VARIABLES ***                  ROW 1 OF 2
COMMAND===>                                                 SCROLL ===> PAGE

Commands:   R - (Remove)    S - (Select)

    Variable     Description                                          Message
    --------     -----------                                          --------
 _  numberio     # of end-of-chain events at channel interface
 _  ticobuch     Time control unit busy at channel (ms)
****************************** BOTTOM OF DATA ********************************
```

Figure 7-60. *Detail Channel Variables panel (SIBDX15)*

```
SIBDX15          *** DETAIL CACHE-EFFECTIVENESS VARIABLES ***      ROW 1 OF 30
 COMMAND===>                                                  SCROLL ===> PAGE

 Commands:   R - (Remove)    S - (Select)

     Variable     Description                                          Message
     --------     -----------                                          --------
  _  bypcareq     Bypass cache request
  _  cacsegct     Number of device cache utilization segments
  _  cadasdxf     Cache to DASD transfer (destages)
⋮
 ****************************** BOTTOM OF DATA ********************************
```

Figure 7-61. *Detail Cache-Effectiveness Variables panel (SIBDX15)*

```
 SIBDX15          *** DETAIL DEVICE-PERFORMANCE VARIABLES ***       ROW 1 OF 20
 COMMAND===>                                                  SCROLL ===> PAGE

 Commands:   R - (Remove)    S - (Select)

     Variable     Description                                          Message
     --------     -----------                                          --------
  _  availtim     Device available time (ms)
  _  busytime     Total physical device utilization (ms)
  _  bytetrak     Bytes per track
⋮
```

Figure 7-62. *Detail Device-Performance Variables panel (SIBDX15)*

a. As each of the detail variables panels appear, select the detail variables to be included in the extracted data by entering beside the variable names one of:

   S Includes the detail variable in data extraction.

   Result: If you press **Enter** at this point, **Selected** appears to the right of the variable name you selected.

   R Removes an earlier selection of the detail variable.

b. Press **Enter**.

   Result: After displaying all the detail variables panels, SVAA returns to the calculated variables panels.

c. On the calculated variables panels, issue the **END** command.

   Result: SVAA proceeds to the next step below.

20. Use the **END** command to return to the SELECT GRAPHS, VARIABLES, OR BOTH panel. On this panel, note that SVAA has inserted an estimated workstation space requirement. This is the number of bytes of storage that SVAA estimates will be needed to store the data for the variables you have specified. If you judge this amount to be too great, go back to step 7 on page 7-53 and type **0** in the **Extraction parameters** field and then repeat steps 16 through 19, specifying smaller time and date ranges or a greater summarization interval.

21. If you are satisfied with the estimated space requirement, use the **END** command to return to the CREATE SAS EXTRACTION PROGRAM panel. Note that there is a message below the **Extracted data format** field that indicates the format you selected.
22. On that panel (Figure 7-50) use the **SUBMIT** command.
23. If the SAS program you specified in step 6 on page 7-52 does not already exist, go to step 25.
24. If the SAS program you specified in step 6 on page 7-52 already exists, specifying this name causes SVAA to use the same name, thus replacing the previous program. To make sure that this is what you want to do, SVAA displays Figure 7-63.

```
SIBDX10                    *** CONFIRM OUTPUT FILE ***
COMMAND===>

The following program already exists:
    'A.B.C(D)'
Do you wish to replace it?    (Y=Yes/N=No)
```

Figure 7-63. *Confirm Output File Name panel (SIBDX10)*

On this panel, SVAA asks whether you wish to replace the previous program. Enter one of:

Y Specifies that you wish to change the name of the SAS program.

Result: SVAA returns to the CREATE SAS EXTRACTION PROGRAM panel for you to specify a new program name.

N Use the same program name, causing the previous program to be overwritten.

Result: SVAA proceeds to the next step.

25. SVAA prepares the output program.

Note: SVAA submits a batch job to extract the data.

Result: SVAA returns you to the SUBSYSTEM REPORTING menu.

Chapter 8. SVAA Subcommands and Commands

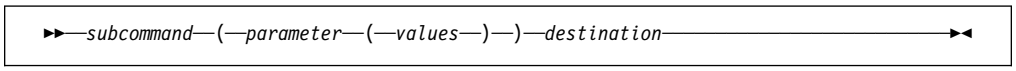
Chapter Summary

The SVAA subcommands can be issued in a variety of ways. Those referred to as "subcommands of SIBBATCH or SIBADMIN" can be executed both interactively and through batch processing. The "System Operator commands" are available to the operator at the OS/390 system console. Many commands and subcommands fall into *both* categories.

Subcommands of SIBBATCH and SIBADMIN

You can issue SVAA subcommands through SIBBATCH and SIBADMIN, and in the SIBSRP initialization routine. This enables you to report on the SVA subsystem from TSO or a batch job.

The general form of SVAA subcommands is shown in the following diagram (the conventions used in the syntax diagrams in this book are explained on page xiv):



where:

subcommand is a subcommand described in this chapter.

parameter is the name of a parameter described in this chapter. (More than one parameter, together with its values, may be included in the subcommand.)

values is one or more of the acceptable values for the specified parameter.

destination is a destination for the output of this particular subcommand to override the destination currently in effect for all SVAA subcommands. For more information, see "Redirecting Subcommand Output" on page 8-10.

Note: You can include a *destination* for output at the end of any SVAA subcommand. However, because you would generally not use it, the destination option is not shown in the subcommand syntax diagrams in this chapter.

Note: You can use the PARM='=PARMLIB=*membername*' parameter on the SIBADMIN or SIBBATCH command to specify the name of the member to be searched for initialization parameters.

Executing Commands from SIBADMIN

In the TSO environment, you can:

- Enter **SIBADMIN** followed by any SVAA subcommand (press **Enter**).
- Enter only **SIBADMIN** and press **Enter**. This establishes the "**SIB**" environment; SVAA returns the **SIB:** prompt, at which you can:
  - Enter an SVAA subcommand. SVAA executes the subcommand and again displays the **SIB:** prompt. Enter another subcommand, or
  - Enter **end** to return to the TSO environment.

See "Issuing Subcommands from TSO" on page 3-6 for more detail.

Executing Commands from SIBBATCH

With SIBBATCH, you can code a set of commands and SVAA subcommands in a sequential data set. Use a SYSIN DD statement in the appropriate JCL to point to the data set containing the commands, and then execute the JCL. Alternatively, you can code the commands in a SYSIN DD * statement and execute the JCL.

You can include comments in sequential data sets following these rules:

1. Enclose comments between matching pairs of /* and */ characters.
2. Comments cannot be nested.
3. Comments can be used only between subcommand keywords and parameters (that is, only where blanks appear in the syntax diagrams in this chapter).

See also “Subcommand Continuation” on page 8-7.

Executing Macros from SIBADMIN and SIBSRP

In SIBADMIN and SIBSRP, you can invoke an SVAA macro to execute a frequently used series of commands, which may include SVAA subcommands. The macro can be executed readily without the need to enter the commands separately. To execute a macro, either:

- Enter **MACRO** *macroname*, where *macroname* is the name of the file that contains the macro (see “MACRO Subcommand” on page 8-33)
- If IMPEX is set on, simply enter the name of the macro in the appropriate environment

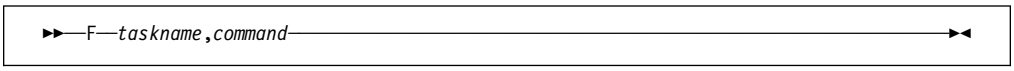
Include comments in an SVAA macro following the rules of the environment.

Using System Operator Commands

Some of the SVAA subcommands are also available to the OS/390 system operator, enabling the operator to interact with the SVA subsystem from a host console. These “subcommands” are marked “***System Operator command***” in their descriptions in this chapter. The system operator commands for Reporter are:

- DISPLAY COLLECTION
- SET COLLECTION (MEMBER)
- SET COLLECTION RESUME
- SET COLLECTION STOP
- SET COLLECTION SUSPEND
- SET COLLECTION SWITCH
- START COLLECTION
- SEND SRP
- START SRP
- STOP SRP

You use a modify (**F**) command to enter SVAA system operator commands. The syntax is:



where *taskname* is the name of an SVAA started task, and *command* is one of the subcommands or commands that are described in this chapter.

Rules for Entering Subcommands and Parameters

Follow these rules when entering SVAA subcommands and their parameters:

- Enter each parameter name only once—SVAA does not allow duplicate parameters.
- Follow each parameter name by the values for that parameter and enclose the set of values in a single pair of parentheses. You must enter a value between the parentheses—for example, DEVPERF() and DEVPERF( ) are not allowed.
- Enter each parameter value only once where multiple values are allowed—SVAA ignores duplicate values.
- Arrange the whole set of parameters in any order.
- Enclose the whole set of parameters within parentheses.
- The SEND subcommands require that the subcommand you are sending (within the parentheses) be enclosed in quotes. If the enclosed subcommand *also* contains quotation marks (that is, nested quotation marks), the outer set of quotes must be different from the inner set. For example: **`SEND SRP("D COLLTN(COLLID('daily'))")`**
- Optionally, for ease of reading, start each new parameter on a new line (you can add additional spaces or use new lines anywhere you want).

Case

You can enter subcommands and parameters in either uppercase or lowercase, but SVAA treats both as uppercase. The *values* you enter for certain parameter *variables*, however, are case sensitive; e.g., "**`a`**" and "**`A`**" are interpreted as different characters.

Whether uppercase and lowercase characters are actually interpreted differently for case-sensitive values is determined by the current setting of the SET CASE subcommand (see "SET CASE" on page 8-67). If you have issued the **`SET CASE(MIXED)`** subcommand, case is preserved; that is, uppercase and lowercase characters are interpreted differently. However, if **`SET CASE(UPPER)`**, the default, is in effect, SVAA interprets all lowercase input as uppercase.

The values you assign to SVA subsystem names, device names and collection identifiers are all case sensitive. A reminder of this fact appears below the "Values:" line wherever one of these parameters is described in this chapter.

To ensure that SVAA correctly interprets the lowercase characters you use for a case-sensitive name, regardless of the current setting of the SET CASE subcommand, enclose the name in matching quotation marks, either single quotes or double quotes. For example: **`'subsysA'`** and **`"Task123"`**. However, in a SEND subcommand, which may involve nested sets of quotation marks, use double quotes for one set and single quotes for the other.

All other parameter values—i.e., those without a note regarding case—are not case sensitive. Such parameters include device addresses and volsers. For example, **`3A0`** and **`3a0`** are the same unit address.

Wildcard Characters

Wildcard characters (* and ?) provide a way for you to specify multiple values for a parameter without typing all of the values individually. You use wildcard characters to indicate positions within a parameter value at which legitimate alphanumeric characters can be substituted according to the following rules:

- The asterisk (*) is the generic wildcard character; it can be replaced by any number of alphanumeric characters or special characters that are acceptable for the specific variable.

  You can separate multiple generic wildcard characters by any number of alphanumeric characters. For example:

  VOLUME(*ROLL*)

  Any volser containing the character sequence **ROLL** is included. Examples are: **PYROLL** and **ROLLFW**

- The question mark (?) is the specific wildcard character; it can be replaced by exactly one alphanumeric character. For example:

  VOLUME(PR?D*) Includes all volumes that start with **PR** and have a **D** in the fourth position.

  VOLUME(PR??D*) Includes all volumes that start with **PR** and have a **D** in the fifth position.

Leading zeros are not significant in matching wildcard specification of device addresses. For example, if you specify **3***, in a device address, devices **3000** to **3FFF**, **300** to **3FF**, **30** to **3F**, and **3** match that specification. However, if you specify **03*** in a device address, only devices **300** to **3FF**, **30** to **3F**, and **3** match that specification.

Ranges

A *range* is a set of sequential values that is defined by the first and last values in the set. Range specifications must adhere to these rules:

- The two values must be of the same data type but have different values.
- The two values must be separated by a colon.
- Except in ranges of units, the two values must contain the same number of characters. Use leading zeroes if necessary.
- Except in unit ranges, both values must contain at least one numeric (decimal) character and the numerics must be in the same position in each value. (Numerics are not required in unit ranges, so AAA:FFF is a valid range of units.)
- The values can be times, single unit addresses, volume serial numbers, or subsystem names. However, it is not necessary for all the values to represent addresses, volsers, or units, or names that exist.

  For example, you can enter a pair of volume serial numbers (volsers), each consisting of 1 to 6 alphanumeric characters, such as:

  P001:P009

- Except in time ranges, the first value must be less than the second.
- Elements in a range cannot contain wildcard characters.
- Multiple ranges within the same parameter must be separated by spaces.

- In time and date ranges, multiple ranges cannot overlap.
- Between two values, only the right-most numeric sequences can differ. All other characters must match.

Note: In ranges of volsers or device names, hexadecimal "digits" are not considered to be numeric. For example, **SYS000:SYS0FF** is not a valid range. Also, **SYS009:SYS015** is a valid range, but does not include **SYS00A** through **SYS00F**.

Examples of valid ranges:

Range	Volsers Defined
A00B0:A00B8	A00B0, A00B1, ... , A00B8
A3B2C5:A3B2C7	A3B2C5, A3B2C6, A3B2C7
2VM31C:2VM49C	2VM31C, 2VM32C, 2VM33C, ... 2VM49C
005:009	005 to 009

Examples of invalid ranges:

Range	Reason Invalid
A20B30:A29B30	The varying sequence (20 ... 29) is not the right-most numeric sequence.
A20B1:A20B99	The elements have differing numbers of digits.
SYS1A:SYSA2	The numerics are not in the same position and the non-numeric portions do not match.

Lists

A *list* consists of two or more parameter values or ranges separated by spaces. The entire list must be enclosed in parentheses. For example:

```
SUBSYS(SUBSYS1 SUBSYS3:SUBSYS6 SUBSYS9)
```

You can use wildcard characters (* and ?) within lists—*if* the parameter is one for which wildcard characters are allowed. For example:

```
EXCLUDEVOLUME(ABC000 ABC* TST??s TST5*)
```

Subcommand Continuation

Because many of the SVAA subcommands are longer than the typical line or record, you may need to continue a subcommand across several records. The method of continuation depends on the facilities of the execution environment:

- Under REXX in TSO, use the REXX method of continuation.
- Within a REXX EXEC, there is a limitation of 512 characters for a statement. You can overcome the limitation by building your command in a series of concatenated clauses.
- In a TSO CLIST, use the CLIST method of continuation.
- In a command at a TSO command prompt, use the TSO continuation.
- The SIBBATCH program uses the same method of continuation as the IDCAMS utility. Except for the last record, each record must have a hyphen or a plus sign as the last non-blank character before the right margin to indicate continuation of the subcommand. You can place the hyphen or plus sign after *any* character (even in the middle of a value) and continue the subcommand on the next line.

- In TSO sessions, SVAA allows a maximum of 32,763 characters in a command or subcommand. SVAA truncates longer commands to this limit and issues the message:

  ```
  SIB0714S   Unexpected system failure of SUBCOMM processing.
  ```

The examples in this chapter use hyphens to show continuation.

Using the Style Prefix

In general,[1] you can qualify all file names in Reporter subcommands with an optional *style* subparameter, as follows:

[*style*:]*filespec*

There can be one or more blanks between *style:* and *filespec*.

The value of **style** can be:

DDN If you use **DDN** for the *style*, *filespec* must refer to a DD statement that is present in either the JCL for the Reporter Data Collection task or the JCL for the SAS step following the SUMM HIST or REPORT subcommand.

Example:

```
SIBADMIN INIT MAIN (OUTFILE (DDN:STKPERF))
```

In the JCL:

```
//STKPERF DD DSN=SRP.MAIN,DISP=SHR
```

Specify a DD name as 1 to 8 alphanumeric or national characters ($ @ #), the first of which cannot be numeric.

DSN If you use **DSN** for the *style*, *filespec* specifies a data set name. Except for the main and alternate OR output files, the data set is dynamically allocated (but the file must already exist).

filespec can contain 1 to 44 alphanumeric or national characters ($ @ #), and 2 special (- {) characters, in “nodes” of 1 to 8 characters separated by periods. Reporter converts lowercase to uppercase. Do not use a numeric (0-9) or special character (- or {) as the first character of any node.

TSO (Allowed only in the TSO environment.) If you use **TSO** for the *style, filespec* specifies a data set name. The data set is dynamically allocated.

filespec can contain 1 to 44 alphanumeric or national characters ($ @ #), and 2 special (- {) characters, in “nodes” of 1 to 8 characters separated by periods. Reporter converts lowercase to uppercase. Do not use a numeric (0-9) or special character (- or {) as the first character of any node.

TSO example:

```
SIBADMIN COLL ORD (OUTFILE (TSO:'STK.PERF.HIST'))
```

Note: Follow standard data set naming conventions; that is, enclose a data set name in quotes if it is fully qualified. (If you do not enclose the data set name in quotes, OS/390 prefixes your user ID to the name, except when you are using the COLL ORD subcommand under SIBADMIN.)

---

[1] Exceptions are SAS file names.

Specifying SAS Data Sets

The following parameters specify SAS data sets:

- HISTIN and HISTOUT parameters of the SUMM HIST subcommand
- RPTSAS, SASDFL, and SASSFL parameters of the REPORT PERF subcommand

Never specify the *style* prefix for SAS files. Specify these data set names as *ddn.sasfilename*, where:

- *ddn* is the DD name (in the JCL for the SAS step) of the SAS library containing the data set (the SAS library name)
- *sasfilename* is the internal file name of the SAS data set.

A *ddname* can be 1 to 8 alphanumeric characters, the first of which cannot be numeric (see Note 2 below).

SAS names you specify must meet these requirements:

- They can be up to 8 characters long.
- The first character must be alphabetic, or an underscore (_). Subsequent characters can be alphabetic, numeric, or underscores. (SAS converts all names to uppercase during processing.)
- Do not use blanks within a SAS name.
- Do not use special characters, except for the underscore. The $, #, and @ characters are allowed only in filerefs. Wildcard characters are not allowed.
- Do not use variable names that are SAS automatic variables or variable list names (for example, _N_, _ERROR_, _NUMERIC_, _ALL_).
- Do not use the names that SAS reserves for particular libraries, including LIBRARY, SASCAT, SASHELP, SASMSG, SASUSER, USER, WORK). Do not use the reserved SAS data set names _NULL_, _DATA_, and _LAST_.

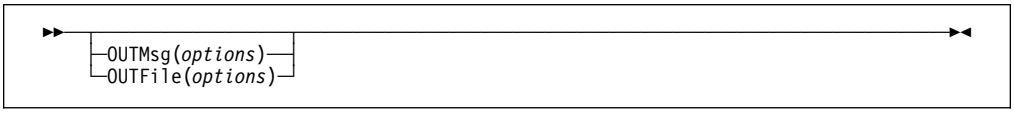
Notes:

1. SAS requires that output SAS library data sets be opened with a disposition of **OLD** in the JCL.
2. In those parameters of REPORT PERF that have a *gens* value, the 8-character limitation on the length of the SAS file name must be reduced by the number of digits in the number of generations. (For example, if 15 generations are specified, the SAS file name is limited to 6 characters.)

Redirecting Subcommand Output

SVAA sends the output of SVAA subcommands to the destination specified in the most recent SET DESTINATION subcommand. For any specific subcommand, you can override this destination by specifying a *destination* at the end of the subcommand (as shown on page 8-3).

The syntax for the *destination* is:



where *options* are any of the subparameters allowed for the OUTFILE or OUTMSG parameters of the SET DESTINATION subcommand (that is, any of *filespec*, DEFAULT, OFF, TERMINAL, or VARIABLE). See “SET DESTINATION” on page 8-70 for explanations of these subparameters.

Note: In the TSO environment, follow standard data set naming conventions; that is, enclose a data set name in quotes if it is fully qualified.

Subcommand Descriptions

The SVAA subcommands are presented in alphabetical order beginning on page 8-11. At the beginning of each subcommand description, the subcommand is identified as a “***Subcommand of SIBBATCH or SIBADMIN***” or as a “***System Operator command***”—or both.

The description of each subcommand consists of:

- An explanation of the command's purpose
- A syntax diagram
- Descriptions of all parameters
- Examples using the subcommand

Parameter Descriptions

Immediately under each parameter name, the parameter is identified as either “***Required***”, “***Optional***”, or “***Conditional***”. The term **Conditional** means that the parameter is neither required nor completely optional but that its use is related to other parameters. An associated statement explains the conditions for its use.

After a brief description of the parameter's purpose, some or all of the following types of information may appear:

Abbreviation: The uppercase portion of the parameter name as shown in the syntax box.

Values: A description of valid input, with any restrictions.

Default value: The default value, if one exists, or “None,” if there is no default.

Examples: Some valid examples of parameter entries.

COLLECT ORDATA Subcommand

* ***Subcommand of SIBBATCH or SIBADMIN***

This subcommand defines the collection parameters for on-request (OR) data collection. See “Collecting On-Request Data” on page 2-4 for more information about OR data.

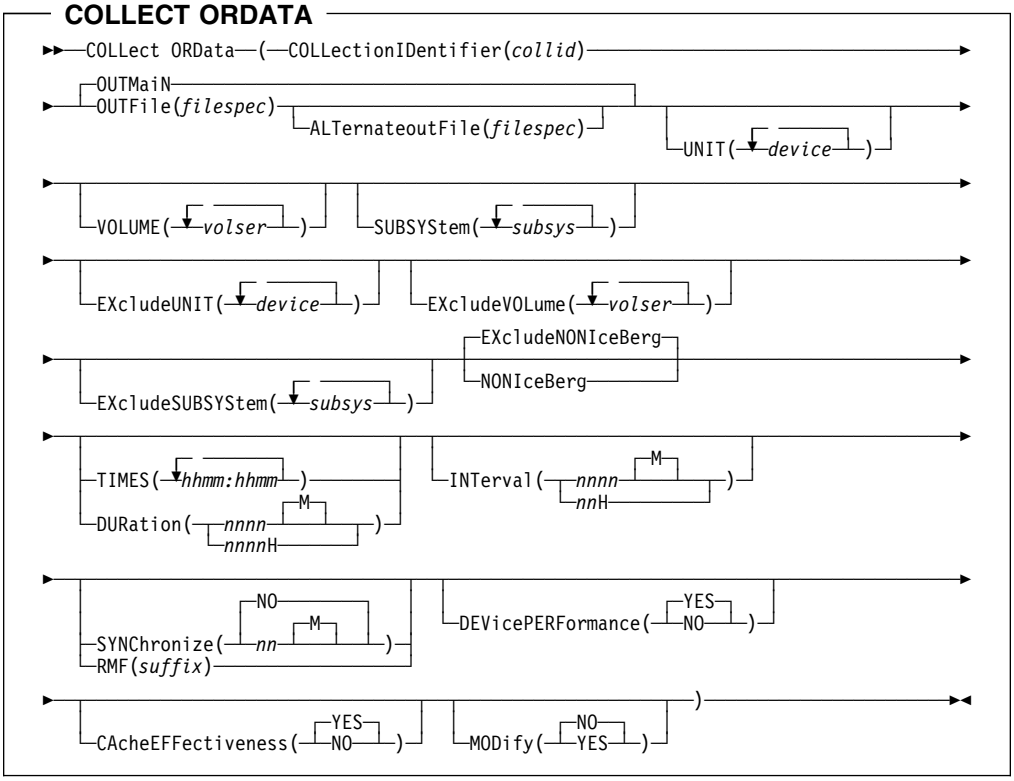
Any number of OR data-collection runs can be active at any given time. Each OR data-collection run is identified by a unique run identifier that you supply. The run identifier must be used when you modify previously submitted COLLECT ORDATA subcommands.

Notes:

1. Performance data is not collected for Power PPRC Bridge devices.
2. PAV Alias devices are excluded from all data collection; thus, performance data is not reported for Aliases.

For OR data collection you can specify:

- A unique identifier for this collection run (COLLECTIONIDENTIFIER parameter).
- Where the collected data is to be stored (OUTMAIN, OUTFILE, and ALTERNATEOUTFILE parameters).
- The devices, volumes, and subsystems for which data is to be collected (UNIT, VOLUME, or SUBSYSTEM parameters).
- The devices, volumes, and subsystems for which data is not to be collected (EXCLUDEUNIT, EXCLUDEVOLUME, or EXCLUDESUBSYSTEM parameters).
- Whether data is to be collected from non-SVA subsystems (NONICEBERG parameter) or not (EXCLUDENONICEBERG parameter).
- Either the ranges of times over which data is to be collected (TIMES parameter), or the length (beginning immediately) of the data collection period (DURATION parameter).
- The length of time between collection instances (INTERVAL parameter).
- The number of minutes after the start of the hour that a collection instance is to occur (SYNCHRONIZE parameter).
- That collection instances are to be synchronized with RMF collection (RMF parameter).
- Whether device-performance data is to be collected (DEVICEPERFORMANCE parameter).
- Whether cache-effectiveness data is to be collected (CACHEEFFECTIVENESS parameter).
- Whether this subcommand modifies (replaces) the current data collection parameters (MODIFY parameter).



See page 8-20 for examples of coding the parameters.

Parameters

The parameters of the COLL ORD subcommand follow, in alphabetical order.

ALTERNATEOUTFILE (filespec)

Optional. If you use this parameter, you must also enter the OUTFILE parameter and you *cannot* enter the OUTMAIN parameter.

This parameter specifies the alternate OR output file for this collection run. If you specify both OUTF and ALTF, the Reporter Data Collection Task can switch to the *alternate output* file if the *output* file fills up.

Abbreviation: **ALTF**

Values: *filespec* specifies the alternate output file. You must use a fully qualified name and a style prefix. See “Using the Style Prefix” on page 8-8 for more information.

Note: In the TSO environment, follow standard data set naming conventions; that is, enclose a data set name in quotes if it is fully qualified.

Default value: If you omit this parameter, there is no alternate OR output file for this collection run.

Notes:

1. If you specify MOD(YES), Reporter ignores the ALTF parameter.
2. You cannot specify the same *filespec* for both OUTF and ALTF.
3. The data set must exist before the collection run is started:
   - To use the DDN: prefix, the data set must have been allocated in the JCL to start the Data Collection Task.

     Example:

     ```
     COLL ORD (COLLID(MONDAY) ALTF(DDN:ALTORD) ...)
     ```

     and, in the JCL for the Data Collection Task:

     ```
     //ALTORD  DD  DSN='COLLOR.MONDAY.ALT9806',DISP=SHR
     ```

   - To use the DSN: prefix, the data set must have been created in some step before the current step (that is, before the SIBBATCH JCL).

CACHEEFFECTIVENESS (YES | NO)

Optional

This parameter specifies whether SVAA is to collect cache-effectiveness data.

Abbreviation: **CAEFF**

Values: **YES** specifies that cache-effectiveness data is to be collected.

NO specifies that cache-effectiveness data is not to be collected.

Default value: **YES**

Note: If you are specifying collection from a non-SVA subsystem, you cannot specify this parameter as CAEFF(NO), because cache-effectiveness data is the only data collected for non-SVA systems.

COLLECTIONIDENTIFIER (collid)

Required

Specifies a unique name for this collection run. This name enables you to have a single output file shared by any number of OR collection runs.

Abbreviation: **COLLID**

Values: *collid* is the name for this collection run. Use up to 16 characters, including a-z, A-Z, 0-9, $, @, #, &, -, +, _, ., or /.

If you use lowercase alphabetic characters and SET CASE(UPPER) is in effect, you must enclose the identifier in matching quotes.

Note: You cannot use a value of **PERFORMANCETRACK**.

Default value: None. You must specify a value.

DEVICEPERFORMANCE (YES | NO)

Optional

This parameter specifies whether Reporter is to collect device-performance data.

Abbreviation: **DEVPERF**

Values: **YES** specifies that performance data is to be collected.

NO specifies that performance data is not to be collected.

Default value: YES

Note: Reporter never collects device-performance data for non-SVA devices.

DURATION (nnnn[M] | nnnnH)

Optional. If you enter this parameter, you cannot enter the TIMES parameter.

This parameter specifies that data collection start immediately and last for the specified duration.

Abbreviation: DUR

Values: *nnnn*[M] specifies a duration of *nnnn* minutes, where *nnnn* can be 1 to 9999. (Leading zeros are optional.) Entering M (the default) is optional.

*nnnn*H specifies a duration of *nnnn* hours, where *nnnn* can be 1 to 9999. (Leading zeros are optional.)

Default value: If you specify neither the DUR nor the TIMES parameter, collection begins immediately and ends at midnight. If you omit the unit (M or H), it defaults to M (minutes).

EXCLUDENONICEBERG

Optional. If you enter this parameter, you cannot enter the NONIB parameter.

This parameter specifies that Reporter is not to collect data from non-SVA disk subsystems. This parameter is the default; Reporter only collects data for non-SVA subsystems if you specify the NONIB parameter, or if the UNIT or VOLUME parameter identifies a non-SVA device.

Abbreviation: EXNONIB

Values: None

Default value: EXNONIB

EXCLUDESUBSYSTEM (subsys)

Optional

This parameter specifies the subsystems to be excluded from data collection.

Abbreviation: EXSUBSYS

Values: *subsys* specifies the names of up to 32 installed SVA subsystems to be excluded. Use exactly the same uppercase or lowercase characters as defined for the subsystem name.

You can use wildcard characters and ranges.

If a subsystem name contains lowercase characters and SET CASE(UPPER) is in effect, you must enclose the name in matching quotes.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter collects data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

EXCLUDEUNIT (device)

Optional

This parameter specifies devices (as they are defined to the host operating system) to be excluded from data collection. You can specify up to 4096 devices on the V2X or up to 1024 on earlier SVAs. Devices to be excluded modify and override the list of devices to be included.

Abbreviation: **EXUNIT**

Values: *device* specifies host device numbers (1 to 4 hexadecimal digits per device). You can pad to the left with zeros. Reporter treats lowercase the same as uppercase.

You can use wildcard characters and ranges.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter collects data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

EXCLUDEVOLUME (volser)

Optional

This parameter specifies the volume serial numbers of devices to be excluded from data collection. You can specify up to 4096 volsers on the V2X or up to 1024 on earlier SVAs. Volumes to be excluded modify and override the list of volumes to be included.

Abbreviation: **EXVOL**

Values: *volser* specifies volume serial numbers of devices to be excluded. SVAA converts all alphabetics in volsers to uppercase.

You can use wildcard characters and ranges.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter collects data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

INTERVAL (nnnnM | nnH)

Optional. If you enter this parameter, you cannot enter the RMF parameter.

This parameter specifies the length of time between data-collection instances.

Abbreviation: **INT**

Values: *nnnn***M** specifies the number of minutes in the collection interval. *nnnn* can be specified as 1 to 1440 minutes (one to four digits, including leading zeros, which are optional). The number must be exactly divisible into 1440 so that collection instances occur at the same times every day. Entering **M**, the default, is optional.

*nn***H** specifies the number of hours in the collection interval. *nn* can be specified as 1 to 24 hours (one to two digits, including a leading

zero, which is optional). The number must be exactly divisible into 24 so that collection instances occur at the same times every day.

Default value: **15M** (15 minutes)

Note: If you specify both the INT and SYNCH parameters, the value of the INTERVAL parameter must be evenly divisible into 60.

For example, you can specify:

```
INT(20) SYNCH(19)
```

but you cannot specify:

INT(18) SYNCH(20) or **INT(2H) SYNCH(20)**

MODIFY (YES | NO)

Optional

This parameter specifies whether this subcommand is a request to modify existing OR data-collection runs specified by the COLLID parameter.

Abbreviation: **MOD**

Values: **YES** is a request to modify an existing data-collection run. Reporter ignores any *filespec* you specify for the OUTF and ALTF parameters.

NO is a request to start a new data-collection run.

Default value: **NO**

Example: See Example 9 on page 4-14.

Note: If you specify **MOD(YES)**, SVAA changes only the values for the parameters you specify in this COLL ORD subcommand, with one exception: If you include any device-selection parameters (subsystems, units, or volumes), their values override and replace the values of the corresponding parameters in the original subcommand.

NONICEBERG

Optional. If you enter this parameter, you cannot enter the EXNONIB parameter.

This parameter specifies that Reporter is to collect data from non-SVA disk subsystems, except for devices excluded by EXUNIT or EXVOL. Only cache statistics available from the 3990 Read Subsystem Data command are produced. There is no non-SVA device selection provided at the control unit or SSID level.

Abbreviation: **NONIB**

Values: None

Default value: **EXNONIB**

Notes:

1. When you are specifying data collection from SIBBATCH, you must send a SET NONIB subcommand to the SVAA address space for each non-SVA device for which you wish to collect data. For example:

   ```
   SEND SRP ('SET NONIB(1E0)')
   ```

2. Performance data is not collected for Power PPRC Bridge devices. If you specify a Power PPRC Bridge device as a non-SVA device using a SET NONIB subcommand, you will receive a SIB1814E message and no data will be collected.

3. PAV Alias devices are excluded from all data collection; thus, performance data is not reported for Aliases.

OUTFILE (filespec)

Optional. If you enter this parameter, you cannot enter the OUTMAIN parameter.

This parameter specifies the primary OR output file for this data-collection run. This file is to be used if data is not to be written to the main logging file.

Abbreviation: **OUTF**

Values: *filespec* identifies the output file. You must use a fully qualified name and a style prefix. See "Using the Style Prefix" on page 8-8 for more information.

Note: In the TSO environment, follow standard data set naming conventions; that is, enclose the data set name in quotes.

Default value: If you don't enter the OUTFILE parameter, OUTMAIN is the default.

Notes:

1. If you specify both OUTF and ALTF, the Reporter Data Collection Task can switch to the *alternate output* file if the *output* file fills up.
2. If you specify MOD(YES), Reporter ignores the OUTF parameter.
3. You cannot specify the same *filespec* for both OUTF and ALTF.
4. Unless you are using SMF for data collection, your site security administrator must have granted update authority to access the OR output file.
5. The data set must have been created before the collection run is started:
   - To use the DDN: prefix, the data set must have been allocated in the JCL to start the Data Collection Task.

     Example:

     ```
     COLL ORD (COLLID(MONDAY) ALTF(DDN:OUTORD) ...)
     ```

     and, in the JCL for the Data Collection Task:

     ```
     //OUTORD  DD  DSN='COLLOR.MONDAY.OUT9806',DISP=SHR
     ```

   - To use the DSN: prefix, the data set must have been allocated in some step before the current step (that is, before the SIBBATCH JCL).

OUTMAIN

Optional. If you enter this parameter, you cannot enter the OUTFILE parameter or ALTF parameter.

This parameter specifies that the output for this collection run is to be directed to the main logging file specified by the OUTF parameter of the INIT MAIN subcommand.

Abbreviation: **OUTMN**

Values: None (keyword parameter only)

Default value: If you enter neither the OUTMAIN parameter nor the OUTFILE parameter, OUTMAIN is the default.

Reporter ignores this parameter if you also specify MOD(YES). This means that you cannot modify the destination for the OR data being collected.

RMF (suffix)

Optional. If you enter this parameter, you cannot enter the INTERVAL parameter or the SYNCHRONIZE parameter.

You can request that Reporter use the same collection interval and synchronization value specified for RMF in the **ERBRMF***nn* member. SVAA searches for this member first in the data sets pointed to by the STKPARMS DD statement; if it is not found there, SVAA searches in SYS1.PARMLIB.

Abbreviation: None.

Values: *suffix* specifies the same value as the *nn* part of **ERBRMF***nn*.

SUBSYSTEM (subsys)

Optional

This parameter specifies one or more SVA subsystem names.

Abbreviation: **SUBSYS**

Values: *subsys* specifies the names of up to 32 installed SVA subsystems. Use exactly the same uppercase and lowercase characters as defined for the subsystem name.

You can use wildcard characters and ranges.

If a subsystem name contains lowercase characters and SET CASE(UPPER) is in effect, you must enclose the name in matching quotes.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter collects data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

SYNCHRONIZE (nn[M] | NO)

Optional. If you enter this parameter, you cannot enter the RMF parameter.

This parameter specifies the minute on which a data-collection instance is to occur. Other instances are synchronized to this instance by the value specified for the collection interval. This parameter has the effect of causing all the collection instances to be offset from the beginning of the time range.

Abbreviation: **SYNC**

Values: *nn*[**M**] specifies data collection is to start *nn* minutes after the hour. *nn* can be one or two digits, 0 to 59. You can use a leading zero.

NO specifies there should be no synchronization of the collection interval. This is equivalent to synchronizing to the start of the next time range, as specified by the TIMES parameter. If you execute SYNC(NO) during a time range, collection starts immediately.

Default value: **NO**

Note: If you specify a value that does not coincide with the beginning of an interval, Reporter offsets the collection interval from the start of a time range so that a collection interval begins at a synchronization time. See the example in “Controlling Intervals” on page 2-5.

TIMES (hhmm:hhmm)

Optional. If you enter this parameter, you cannot enter the DURATION parameter.

This parameter specifies the start-stop time-range pairs for data collection. On-request data collection occurs for the 24 hours following the subcommand being issued and during the time ranges specified.

Values: *hhmm:hhmm* specifies up to 16 pairs of times for which data collection is to be started and stopped (in hours and minutes, using the 24-hour clock). The first value in the pair is the starting time and must be between 0000 and 2359; the second is the stopping time and must be between 0001 and 2400. Separate the two times by a colon; separate multiple ranges by spaces. If the stopping time is earlier than the starting time, collection spans midnight. You cannot specify overlapping time-range pairs. You can specify ranges in any order.

Default value: If you do not specify the TIMES or DUR parameters, Reporter collects data all day. Specifying 0000:2400 also results in all-day data collection.

Note: See “Specifying Time Ranges for Collection” on page 4-5 for examples and more information.

UNIT (device)

Optional

This parameter specifies devices (as they are defined to the host operating system) for which data is to be collected. You can specify up to 4096 devices on the V2X or up to 1024 on earlier SVAs.

Values: *device* specifies host device numbers (1 to 4 hexadecimal digits per device). You can pad to the left with zeros. Reporter treats lowercase the same as uppercase.

You can use wildcard characters and ranges.

Examples: **`UNIT(2D4 2D5 3A0:3BF)`** requests data collection for devices **`2D4`**, **`2D5`** and all devices with addresses in the range **`3A0`** to **`3BF`**.

`UNIT(3*)` requests data collection for all devices with the first digit of the address equal to 3.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter collects data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

VOLUME (volser)

Optional

This parameter specifies the volume serial numbers of devices for which data is to be collected. You can specify up to 4096 volsers on the V2X or up to 1024 on earlier SVAs.

Abbreviation: **`VOL`**

Values: *volser* specifies volume serial numbers of devices to be included. SVAA converts all alphabetics in volsers to uppercase.

You can use wildcard characters and ranges.

Example: **VOL(PROD01:PROD25)** requests data collection for all devices with volume serial number **PROD01** through **PROD25**.

VOL(PR*) requests data collection for all devices with volume serial numbers starting with **PR**, while **VOL(PR?D*)** requests inclusion of all devices with volume serial numbers starting with **PR** and with a fourth byte of **D**.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter collects data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

Examples of COLLECT ORDATA

Example 1. Starting on-request data collection

```
COLL ORD (-
    COLLID(COLLCT220) )
```

Reporter requests on-request data collection with the default values for all parameters. The collection identifier is to be **COLLCT220**. Data is collected for all subsystems and devices from the time the subcommand is issued until midnight, without any synchronization of the collection interval. Both device-performance and cache-effectiveness data is collected. Data is directed to the main logging file.

Example 2. Specifying collection times for specific functional devices

```
COLL ORD (-
    collid(collct220)-
    Times(1030:1115)-
    Unit(220 222 223)-
    Int(5M)-
    Sync(NO) )
```

Data is to be collected from 10:30 a.m until 11:15 a.m., for units 220, 222, and 223. Data is collected at five-minute intervals during the specified times. Both device performance and cache effectiveness data is collected. The collection run terminates at 11:15. Data is directed to the main logging file.

Example 3. Specifying a duration and a logging file

```
COLL ORD (-
    COLLID(COLLECTA)-
    DUR(2H)-
    OUTF(DDN:OUTFIL1)-
    int(20M)-
    sync(10M)-
    SUBSYS(SUBSYSA SUBSYSB) )
```

Data is to be collected starting immediately and for the next two hours. The collection identifier is COLLECTA—output records are marked with that identifier. Data is written to a sequential logging file pointed to by the OUTFIL1 DD statement in the JCL. Data is to be collected for all devices in subsystems SUBSYSA and

SUBSYSB. The collection interval is 20 minutes and the intervals start at ten minutes, thirty minutes, and fifty minutes after the hour.

COLLECT PTDATA Subcommand

**** Subcommand of SIBBATCH or SIBADMIN***

The COLLECT PTDATA subcommand defines performance-tracking (PT) data collection. The abbreviation is COLL PT. PT data is not collected unless you use this subcommand.

Only one PT data-collection run can be active at any given time for a subsystem.

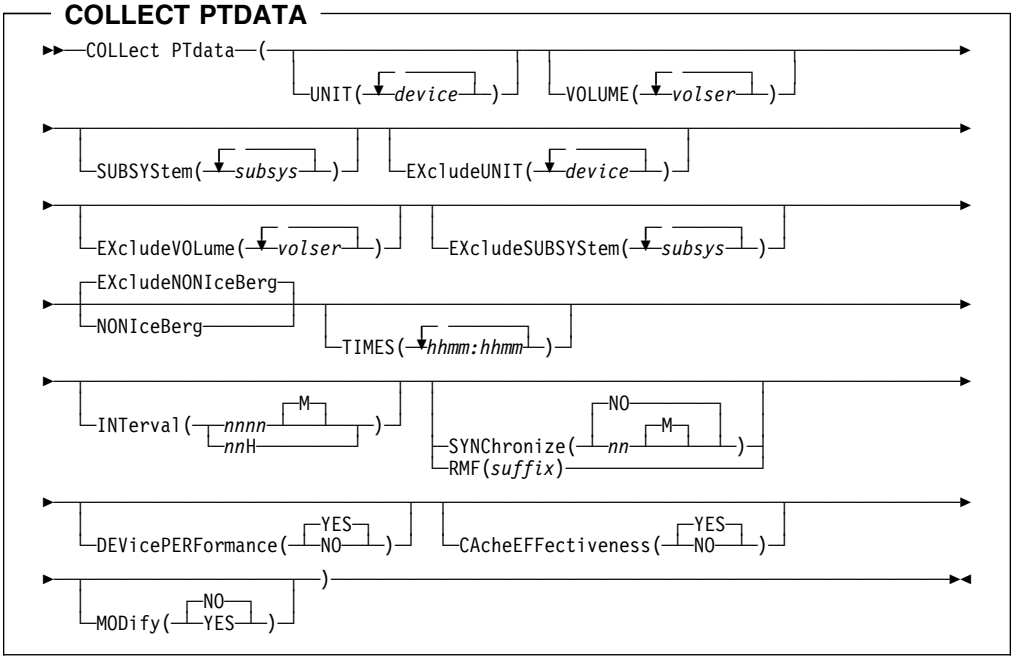
Notes:

1. Performance data is not collected for Power PPRC Bridge devices.
2. PAV Alias devices are excluded from all data collection; thus, performance data is not reported for Aliases.

For PT data collection you can specify:

- The devices, volumes, and subsystems for which data is to be collected (UNIT, VOLUME, or SUBSYSTEM parameters).
- The devices, volumes, and subsystems for which data is not to be collected (EXCLUDEUNIT, EXCLUDEVOLUME, or EXCLUDESUBSYSTEM parameters).
- Whether data is to be collected from non-SVA subsystems (NONICEBERG parameter) or not (EXCLUDENONICEBERG parameter).
- The ranges of times over which data is to be collected (TIMES parameter).
- The length of time between collection instances (INTERVAL parameter).
- The number of minutes after the start of a time range that a collection instance is to occur (SYNCHRONIZE parameter).
- That collection instances are to be synchronized with RMF collection (RMF parameter).
- Whether device-performance data is to be collected (DEVICEPERFORMANCE parameter).
- Whether cache-effectiveness data is to be collected (CACHEEFFECTIVENESS parameter).
- Whether this subcommand modifies (replaces) the current data collection parameters (MODIFY parameter).

Note: SVAA collects PT data in the logging files defined with the INIT MAIN subcommand.



Parameters

All of the parameters of the COLL PT subcommand are optional. See page 8-28 for examples of coding the parameters. The parameters are in alphabetical order.

If you omit all the parameters, all the default values are used (see Example 1 on page 8-28.)

CACHEEFFECTIVENESS (YES | NO)

Optional

This parameter specifies whether SVAA is to collect cache-effectiveness data.

Abbreviation: **CAEFF**

Values: **YES** specifies that cache-effectiveness data is to be collected.

NO specifies that cache-effectiveness data is not to be collected.

Default value: **YES**

Note: If you are specifying collection from a non-SVA subsystem, you cannot specify this parameter as CAEFF(NO), because cache-effectiveness data is the only data collected for non-SVA systems.

DEVICEPERFORMANCE (YES | NO)

Optional

This parameter specifies whether Reporter is to collect device-performance data.

Abbreviation: **DEVPERF**

Values: **YES** specifies that device-performance data is to be collected.

NO specifies that device-performance data is not to be collected.

Default value: **YES**

Note: Reporter never collects device-performance data for non-SVA devices.

EXCLUDENONICEBERG

Optional. If you enter this parameter, you cannot enter the NONIB parameter.

This parameter specifies that Reporter is not to collect data from non-SVA disk subsystems. This parameter is the default; Reporter only collects data for non-SVA subsystems if you specify the NONIB parameter, or if the UNIT or VOLUME parameter identifies a non-SVA device.

Abbreviation: **EXNONIB**

Values: None

Default value: **EXNONIB**

EXCLUDESUBSYSTEM (subsys)

Optional

This parameter specifies the subsystems to be excluded from data collection.

Abbreviation: **EXSUBSYS**

Values: *subsys* specifies the names of up to 32 installed SVA subsystems to be excluded. Use exactly the same uppercase or lowercase characters as defined for the subsystem name.

You can use wildcard characters and ranges.

If a subsystem name contains lowercase characters and SET CASE(UPPER) is in effect, you must enclose the name in matching quotes.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter collects data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

EXCLUDEUNIT (device)

Optional

This parameter specifies devices (as they are defined to the host operating system) to be excluded from data collection. You can specify up to 4096 devices on the V2X or up to 1024 on earlier SVAs. Devices to be excluded modify and override the list of devices to be included.

Abbreviation: **EXUNIT**

Values: *device* specifies host device numbers (1 to 4 hexadecimal digits per device). You can pad to the left with zeros. Reporter treats lowercase the same as uppercase.

You can use wildcard characters and ranges.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter collects data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

EXCLUDEVOLUME (volser)

Optional

This parameter specifies the volume serial numbers of devices to be excluded from data collection. You can specify up to 4096 volsers on the V2X or up to 1024 on earlier SVAs. Volumes to be excluded modify and override the list of volumes to be included.

Abbreviation: **EXVOL**

Values: *volser* specifies volume serial numbers of devices to be excluded. SVAA converts all alphabetics in volsers to uppercase.

You can use wildcard characters and ranges.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter collects data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

INTERVAL (nnnnM | nnH)

Optional. If you enter this parameter, you cannot enter the RMF parameter.

This parameter specifies the length of time between data-collection instances.

Abbreviation: **INT**

Values: *nnnn***M** specifies the number of minutes in the collection interval. *nnnn* can be specified as 1 to 1440 minutes (one to four digits, including leading zeros, which are optional). The number must be exactly divisible into 1440 so that collection instances occur at the same times every day. Entering **M**, the default, is optional.

*nn***H** specifies the number of hours in the collection interval. *nn* can be specified as 1 to 24 hours (one to two digits, including a leading zero, which is optional). The number must be exactly divisible into 24 so that collection instances occur at the same times every day.

Default value: **15M** (15 minutes)

Note: If you specify both the INT and SYNCH parameters, the value of the INTERVAL parameter must be evenly divisible into 60.

For example, you can specify:

```
INT(20) SYNCH(19)
```

but you cannot specify:

INT(18) SYNCH(20) or **INT(2H) SYNCH(20)**

MODIFY (YES | NO)

Optional

This parameter specifies whether the current PT data-collection run is to be modified.

Abbreviation: **MOD**

Values: **YES** is a request to modify an existing data-collection run.

NO is a request to start a new data-collection run.

Default value: **NO**

Example: See Example 9 on page 4-14.

Note: If you specify **MOD(YES)**, SVAA changes only the values of the parameters you specify in this COLL PT subcommand, with one exception: If you include any device-selection parameters (subsystems, units, or volumes), their values override and replace the values of the corresponding parameters in the original subcommand.

NONICEBERG

Optional. If you enter this parameter, you cannot enter the EXNONIB parameter.

This parameter specifies that Reporter is to collect data from non-SVA disk subsystems, except for devices excluded by EXUNIT or EXVOL. Only cache statistics available from the 3990 Read Subsystem Data command are produced. There is no non-SVA device selection provided at the control unit or SSID level.

Abbreviation: **NONIB**

Values: None

Default value: **EXNONIB**

Notes:

1. When you are specifying data collection from SIBBATCH, you must send a SET NONIB subcommand to the SVAA address space for each non-SVA device for which you wish to collect data. For example:

   ```
   SEND SRP ('SET NONIB(1E0)')
   ```

2. Performance data is not collected for Power PPRC Bridge devices. If you specify a Power PPRC Bridge device as a non-SVA device using a SET NONIB subcommand, you will receive a SIB1814E message and no data will be collected.

3. PAV Alias devices are excluded from all data collection; thus, performance data is not reported for Aliases.

RMF (suffix)

Optional. If you enter this parameter, you cannot enter the INTERVAL parameter or the SYNCHRONIZE parameter.

You can request that Reporter use the same collection interval and synchronization value specified for RMF in the **ERBRMF***nn* member. SVAA searches for this member first in the data sets pointed to by the STKPARMS DD statement; if it is not found there, SVAA searches in SYS1.PARMLIB.

Abbreviation: None.

Values: *suffix* specifies the same value as the *nn* part of **ERBRMF***nn*.

SUBSYSTEM (subsys)

Optional

This parameter specifies one or more SVA subsystem names.

Abbreviation: **SUBSYS**

Values: *subsys* specifies the names of up to 32 installed SVA subsystems. Use exactly the same uppercase and lowercase characters as defined for the subsystem name.

You can use wildcard characters and ranges.

If a subsystem name contains lowercase characters and SET CASE(UPPER) is in effect, you must enclose the name in matching quotes.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter collects data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

SYNCHRONIZE (nn[M] | NO)

Optional. If you enter this parameter, you cannot enter the RMF parameter.

This parameter specifies the minute on which a data-collection instance is to occur. Other instances are synchronized to this instance by the value specified for the collection interval. This parameter has the effect of causing all the collection instances to be offset from the beginning of the time range.

Abbreviation: **SYNC**

Values: *nn*[**M**] specifies data collection is to start *nn* minutes after the hour. *nn* can be one or two digits, 0 to 59. You can use a leading zero.

NO specifies there should be no synchronization of the collection interval. This is equivalent to synchronizing to the start of the next time range, as specified by the TIMES parameter. If you execute SYNC(NO) during a time range, collection starts immediately.

Default value: **NO**

Note: If you specify a value that does not coincide with the beginning of an interval, Reporter offsets the collection interval from the start of a time range so that a collection interval begins at a synchronization time. See the example in “Controlling Intervals” on page 2-5.

TIMES (hhmm:hhmm)

Optional

This parameter specifies the start-stop time-range pairs for data collection. SVAA collects data every day during the specified times, until collection is stopped, modified, or suspended.

Values: *hhmm:hhmm* specifies up to 16 pairs of times for which data collection is to be started and stopped (in hours and minutes, using the 24-hour clock). The first value in the pair is the starting time and must be between 0000 and 2359; the second is the stopping time and must be between 0001 and 2400. Separate the two

times by a colon; separate multiple ranges by spaces. If the stopping time is earlier than the starting time, collection spans midnight. You cannot specify overlapping time-range pairs. You can specify ranges in any order.

Default value: If you do not specify the TIMES parameter, Reporter collects data all day. Specifying 0000:2400 also results in all-day collection.

Note: See "Specifying Time Ranges for Collection" on page 4-5 for examples and more information.

UNIT (device)

Optional

This parameter specifies devices (as they are defined to the host operating system) for which data is to be collected. You can specify up to 4096 devices on the V2X or up to 1024 on earlier SVAs.

Values: *device* specifies host device numbers (1 to 4 hexadecimal digits per device). You can pad to the left with zeros. Reporter treats lowercase the same as uppercase.

You can use wildcard characters and ranges.

Examples: **UNIT(2D4 2D5 3A0:3BF)** requests data collection for devices **2D4, 2D5** and all devices with addresses in the range **3A0** to **3BF**.

UNIT(3*) requests data collection for all devices with the first digit of the address equal to 3.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter collects data for all devices on all SVA subsystems.
2. See "Specifying Devices for Data Collection" on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

VOLUME (volser)

Optional

This parameter specifies the volume serial numbers of devices for which data is to be collected. You can specify up to 4096 volsers on the V2X or up to 1024 on earlier SVAs.

Abbreviation: **VOL**

Values: *volser* specifies volume serial numbers of devices to be included. SVAA converts all alphabetics in volsers to uppercase.

You can use wildcard characters and ranges.

Example: **VOL(PROD01:PROD25)** requests data collection for all devices with volume serial number **PROD01** through **PROD25**.

VOL(PR*) requests data collection for all devices with volume serial numbers starting with **PR**, while **VOL(PR?D*)** requests inclusion of all devices with volume serial numbers starting with **PR** and with a fourth byte of **D**.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter collects data for all devices on all SVA subsystems.
2. See "Specifying Devices for Data Collection" on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

Examples of COLLECT PTDATA

Example 1. Collecting PT data with all defaults for all subsystems

```
COLL PT
```

PT data is to be collected from the time the subcommand is issued until midnight for all devices in all SVA subsystems. Reporter collects data for both device performance and cache effectiveness data. The collection interval is 15 minutes, and collection is not synchronized.

Example 2. Collecting performance data from a specific subsystem with RMF synchronization

```
COLL PT(SUBSYS(SUBSYSA) RMF(01) CAEFF(NO))
```

PT data is to be collected form the time the subcommand is issued until midnight for all devices in subsystem SUBSYSA and no other devices. Only device performance data is collected. The collection interval and synchronization values to be used are to be taken from SYS1.PARMLIB member ERBRMF01.

Example 3. Collecting data for specific subsystems during certain time ranges

```
COLL PT-
    (-
    (TIMES(0800:1200 1300:1800)-
    SUBSYS(SUBSYSA SUBSYSB)-
    SYNC(0M)-
    EXVOL(VOL001)-
    EXUNIT(321 322 328)-
    )
```

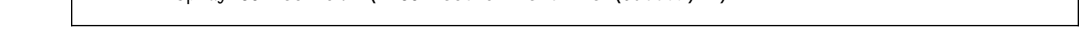
PT data is to be collected from 8 a.m. until noon, and from 1 p.m. until 6 p.m. each day. Data is to be collected for all devices in subsystems SUBSYSA and SUBSYSB, except for volume **VOL001** and devices with addresses **321, 322,** and **328**. The collection interval is 15 minutes, and collection is synchronized on the hour.

DISPLAY COLLECTION Subcommand

* ***Subcommand of SIBADMIN or SIBBATCH***
* ***System Operator command***

You use the DISPLAY COLLECTION subcommand to display information about all current collection runs or about a specific run.

DISPLAY COLLECTION



Parameters

COLLECTIONIDENTIFIER (collid)

Optional

Specifies the unique name of a single collection run to be displayed.

Abbreviation: **COLLID**

Values: For OR data collection, *collid* is the name specified by the COLLID parameter of the COLL ORD subcommand that created the run. For PT data, the *collid* value is always PERFORMANCETRACK.

Default: Reporter displays an abbreviated listing of all collection runs.

Examples of DISPLAY COLLECTION

Example 1. (no collection ID specified)

```
D COLLTN
```

Result:

```
Logging Collection ID  PERFORMANCETRACK  START:   08JUN1999 00:00:00
Collection Type:    PT                   SUSPEND: 12JUN1999 00:00:00
```

Two lines are generated for each collection run. In this case, the only active collection run was a PT run.

Example 2. (with collection ID specified)

```
D COLLTN (COLLID (MONDAYMORN))
```

Result:

```
SRP Collection ID   MONDAYMORN         START:     26Jun1999 21:24:39
Collection Type:    OR
Output to:          TH990626 VM120W A1  (MAINLOG)
Alternate File:     TH990626 VM120W C1
Collection Intrval: 0002M   Synchronize: NO      RMF synchronization: NO
Collection Data:    DEVICEPERFORMANCE   CACHEEFFECTIVENESS
Record count:       000000259  Next Collect Time: 26Jun1999 21:28:00
Collection Times:   00:00-24:00
Subsystems:         IXFPIST
```

Reporter displays additional lines when you specify the COLLID parameter. In general, the display includes the names of the subsystems, volumes, and units from which SVAA collected the data, as specified with the COLL ORD or COLL PT subcommand that created the collection run.

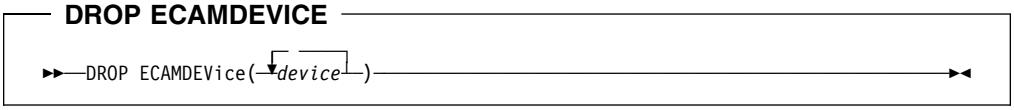
DROP Subcommands

The DROP subcommands release devices that are being used by the host operating system to communicate with either an SVA subsystem or a non-SVA subsystem.

DROP ECAMDEVICE

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand specifies devices (as defined to the host) to be dropped from the list of active ECAM devices.



Abbreviation: **DROP ECAMDEV**

Values: *device* is a host device number (1 to 4 hexadecimal digits).

You can use ranges and lists, but not wildcard characters.

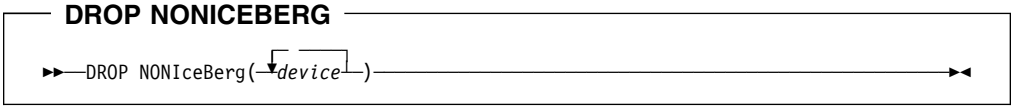
Default value: None

Example: **DROP ECAMDEV(2AE 730 1D)**

DROP NONICEBERG

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand specifies devices (as defined to the host) to be dropped from the list of active non-SVA communication devices (that is, devices over which statistics are obtained from non-SVA subsystems).



Abbreviation: **DROP NONIB**

Values: *device* is a host device number (1 to 4 hexadecimal digits).

You can use ranges and lists, but not wildcard characters.

Default value: None

Example: **DROP NONIB(720 1ED)**

INITIALIZE MAINLOG Subcommand

**** Subcommand of SIBBATCH, SIBADMIN or SIBSRP***

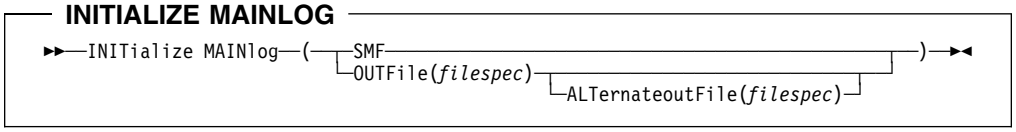
Use the INITIALIZE MAINLOG subcommand to initialize the Reporter Data Collection Task. The abbreviation is INIT MAIN.

This subcommand should be in the initialization subcommands for Reporter in the startup PARMLIB member for the Data Collection Task.

In initializing the data collection task, you can specify:

- That collected data is to be sent to SMF (SMF parameter).
- The primary file where the collected data is to be stored (OUTFILE parameter).
- An alternative file where collected data is to be stored (ALTERNATEOUTFILE parameter).

INITIALIZE MAINLOG



Parameters

One of the SMF or OUTF parameters is required. See "Examples of INIT MAIN" on page 8-33 for examples of coding the parameters.

ALTERNATEOUTFILE (filespec)

Optional. Do not specify this parameter without also specifying the OUTFILE parameter.

This parameter specifies the alternate logging file for collected PT data. If you specify both OUTF and ALTF, the Reporter Data Collection Task can switch to the *alternate output* file if the *output* file fills up.

Abbreviation: **ALTF**

Value: *filespec* specifies the alternate logging file. You can use the *style* prefix (see "Using the Style Prefix" on page 8-8 for more information).

Default value: No alternate logging file is available.

Notes:

1. You cannot specify the same *filespec* for both OUTF and ALTF.
2. The data set must have been allocated before the Data Collection Task is started:
   - To use the DDN: prefix, the data set must have been allocated in the JCL for the SVAA started task.

     Example:

     ```
     INIT MAIN (ALTF(DDN:MAINLOG))
     ```

     and, in the JCL for the SVAA started task:

     ```
     //MAINLOG  DD  DSN=IBM.MAIN,DISP=SHR
     ```

- To use the DSN: prefix, the data set must have been created in some step before the current step.

OUTFILE (filespec)

Conditional. You must enter either this parameter or the SMF parameter, but not both.

This parameter specifies the name of the main logging file for collected PT data. If you specify both OUTF and ALTF, the Reporter Data Collection Task can switch to the *alternate output* file if the *output* file fills up.

Abbreviation: **OUTF**

Value: *filespec* specifies the primary logging file. You can use the *style* prefix (see "Using the Style Prefix" on page 8-8 for more information).

Default value: None

Notes:

1. You cannot specify the same *filespec* for both OUTF and ALTF.
2. Your site security administrator must have granted update authority to the SVAA started task for access to the main logging file.
3. The data set must exist before the Data Collection Task is started:
   - To use the DDN: prefix, the data set must have been allocated in the JCL for the SVAA started task.

     Example:

     ```
     INIT MAIN (OUTF(DDN:MAINLOG))
     ```

     and, in the JCL for the SVAA started task:

     ```
     //MAINLOG  DD  DSN=IBM.MAIN,DISP=SHR
     ```

   - To use the DSN: prefix, the data set must have been created in some step before the current step.

SMF

Conditional. You must enter either this parameter or the OUTFILE parameter, but not both.

This parameter specifies that Reporter is to write SVA subsystem performance data records to SMF, using the default SMF record number supplied in the SVAA initialization parameters.

Abbreviation: None

Values: None (keyword parameter only)

Note: If you wish to collect data both in SMF and in flat files, specify COLL PT with the SMF parameter and COLL ORD with the same parameters as the COLL PT subcommand, except for specifying the output file and a very large value for the DURATION parameter.

Examples of INIT MAIN

Example 1. Output is to SMF

```
INIT MAIN(SMF)
```

Reporter is to write output to SMF.

Example 2. Output is to two logging files

```
INIT MAIN (-
    (OUTF(PROD.HISTFIL1)-
     ALTF(PROD.HISTFIL2))
```

Reporter is to write output to two alternating sequential logging files, called PROD.HISTFIL1 and PROD.HISTFIL2. If the default file-switching exit is used, the collection task automatically switches to the alternate file when the current file fills up.

Example 3. Output is to a single logging file

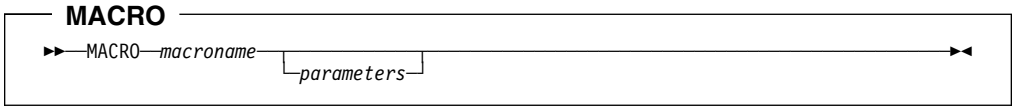
```
INIT MAIN(OUTF(DDN:MAINLOG))
```

Reporter is to write output to a single sequential logging file pointed to by the **MAINLOG DD** statement.

MACRO Subcommand

***** ***Subcommand of SIBADMIN***

The MACRO subcommand executes an SVAA macro.



The *macroname* parameter specifies the name of the macro to be invoked.

In TSO, the macro is a CLIST or a REXX exec that has a member name of *macroname* and is located in a data set found in the SYSPROC DD statement.

QUERY Subcommands

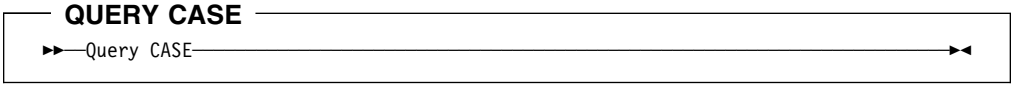
QUERY subcommands request the display of information about SVAA.

You use some QUERY subcommands to query a value set by a SET subcommand, by a profile during initialization, or by default. SVAA responds to these QUERY subcommands with a statement of the source of the setting, followed by the current value of the setting. SVAA responds to other QUERY subcommands with data in numbered SVAA messages.

QUERY CASE

* ***Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand asks whether values of parameters that normally allow lowercase characters are automatically translated to uppercase.



Abbreviation: **Q CASE**

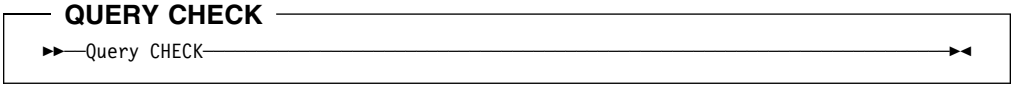
Response: CASE last set by *source*: UPPER or MIXED

where *source* is one of **parmlib, profile, subcommand,** or **default**.

QUERY CHECK

* ***Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand asks whether SVAA subcommand syntax-checking or simulation—or neither—is being performed.



Abbreviation: **Q CHECK**

Response: CHECK last set by *source*: NONE or SCAN or SIMU

where *source* is one of **profile, subcommand,** or **default**,

NONE indicates that neither syntax-checking nor simulation is in effect.

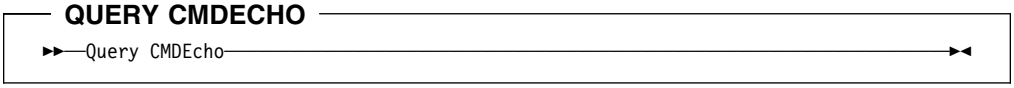
SCAN indicates that syntax-checking is in effect.

SIMU indicates that simulation is in effect.

QUERY CMDECHO

* ***Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand asks whether SVAA subcommands are displayed at the destination specified by **SET DEST OUTMSG** before being executed.



Abbreviation: **Q CMDE**

Response: CMDECHO last set by *source*: ON or OFF

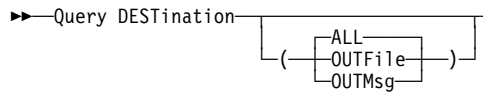
where *source* is one of **profile, subcommand,** or **default**.

QUERY DESTINATION

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand asks for the current destination for reports and messages, as set by default during SVAA initialization, by execution of a profile, or by a SET DESTINATION subcommand.

QUERY DESTINATION



Abbreviation: **Q DEST**

Values: **ALL** asks for the destinations for both report and message output.

OUTF asks for the destination for configuration reports (for example, LISTCFG subcommand output).

OUTM asks for the destination for message output.

Default value: **ALL**

Response: OUTMSG last set by *source: destination*
OUTFILE last set by *source: destination*

where *source* is one of **profile, subcommand,** or **default**, and *destination* is one of:

TERMINAL—output goes to the user's terminal.

filename filetype—the name of a file to which report or message output is being sent.

EXEC VARIABLES—output is directed to EXEC stem variables

NULL DESTINATION—output is suppressed

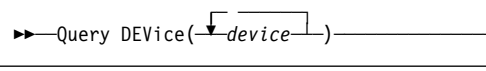
QUERY DEVICE

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand asks for SVA subsystem information about a host device.

Note: You cannot use this subcommand to display information about PAV Alias devices or Power PPRC Bridge devices. Use the DISPLAY DEVICE or LISTCFG DEVICE subcommands instead.

QUERY DEVICE



Abbreviation: **Q DEV**

Values: *device* is a host device number (1 to 4 hexadecimal digits).

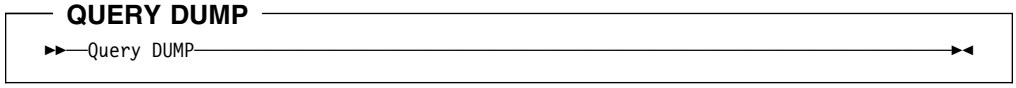
You can indicate devices with both lists and ranges.

Response: SVAA messages SIB1868I and SIB1869I.

QUERY DUMP

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand asks whether a dump is to be created if SVAA terminates abnormally, and, if so, the ID of the TSO user to be notified.



Abbreviation: **Q DUMP**

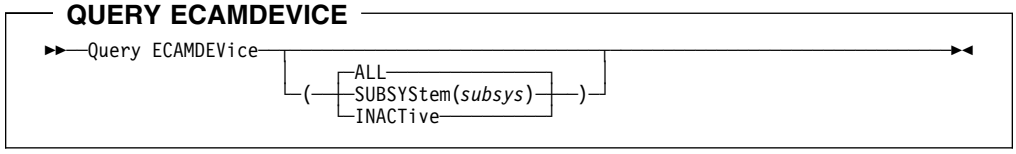
Response: DUMP last set by *source*: ON or OFF DUMPID=* or id

where *source* is one of **profile, subcommand,** or **default**.

QUERY ECAMDEVICE

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand asks for the addresses of the ECAM devices defined for the specified subsystems (or all subsystems), or for the addresses of devices *not* currently active as ECAM communications devices.



Abbreviation: **Q ECAMDEV**

Values: **ALL** asks for the addresses of *all* devices defined as ECAM communications devices on *all* SVA subsystems.

SUBSYS(*subsys*) asks for the addresses of all ECAM communications devices for subsystem *subsys*.

INACT asks for the addresses of devices for which a SET ECAMDEVICE subcommand was issued when the device was offline.

Default value: **ALL**

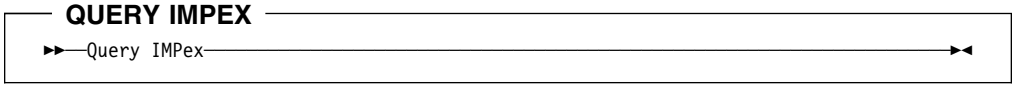
Response: SVAA messages SIB1820I and SIB1821I.

QUERY IMPEX

*** *Subcommand of SIBADMIN, or SIBSRP***

This subcommand asks for the status for executing SVAA macros. If IMPEX is on, SVAA macros are executed (if they exist) before treating the subcommand as an SVAA subcommand.

QUERY IMPEX



Abbreviation: **Q IMP**

Response: IMPEX last set by *source*: ON or OFF

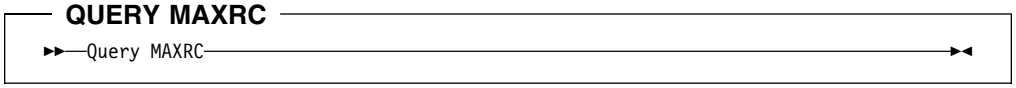
where *source* is one of **profile, subcommand,** or **default**.

QUERY MAXRC

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand asks for the current MAXRC value (the maximum severity return code).

QUERY MAXRC



Abbreviation: **Q MAXRC**

Response: MAXRC last set by *source*: *nn*

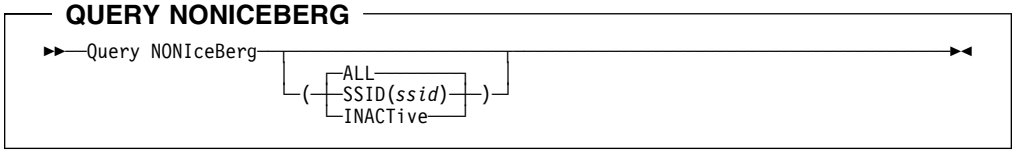
where *source* is one of **profile, subcommand,** or **default** and *nn* is the maximum return code.

QUERY NONICEBERG

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand asks for the addresses of devices over which non-SVA subsystem statistics are obtained.

QUERY NONICEBERG



Abbreviation: **Q NONIB**

Values: **ALL** asks for the addresses of *all* devices over which statistics are obtained for non-SVA subsystems.

SSID(*ssid*) asks for the addresses of all devices on subsystem *ssid*.

INACT asks for the addresses of devices that were offline when the SET NONICEBERG subcommand was issued.

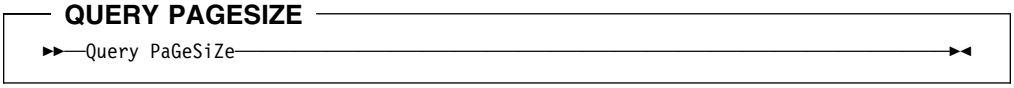
Default value: **ALL**

Response: SVAA messages SIB1822I and SIB1823I.

QUERY PAGESIZE

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand asks for the currently specified number of lines per page for SVAA reports.



Abbreviation: **Q PGSZ**

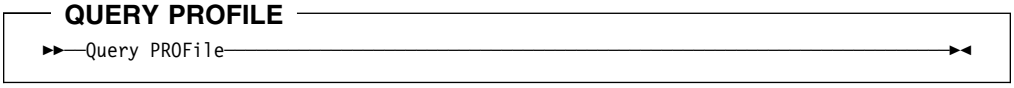
Response: PAGESIZE last set by *source*: *nn*

where *source* is one of **profile, subcommand,** or **default**, and *nn* is the number of lines per page.

QUERY PROFILE

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand asks for the name of the SVAA profile macro.



Abbreviation: **Q PROF**

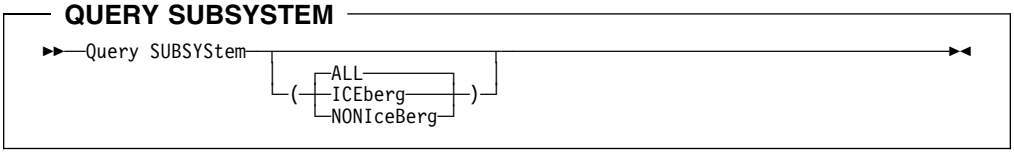
Response: PROFILE last set by *source*

where *source* is either **PROFSIBA** or a *filename*.

QUERY SUBSYSTEM

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand asks for information about SVA and non-SVA subsystems.



Abbreviation: **Q SUBSYS**

Values: **ALL** asks for information about all SVA and non-SVA subsystems.

ICE asks for information about SVA subsystems only.

NONIB asks for information about non-SVA subsystems only.

Default value: **ALL**

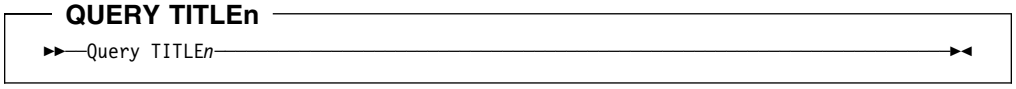
Response: **SVA** *subsys* **serial#** *serial* **has** *numdev* **ECAM devices.**

where *subsys* is the subsystem name, *serial* is the serial number of the subsystem, and *numdev* is the number of ECAM devices defined for the SVA subsystem.

QUERY TITLEn

*** *Subcommand of SIBBATCH or SIBADMIN***

This subcommand asks for the text of a report title line—either **Title1** (line 1), **Title2** (line 2), or **Title3** (line 3).



Abbreviation: **Q TITLEn** (where *n* is **1, 2, *or* 3**)

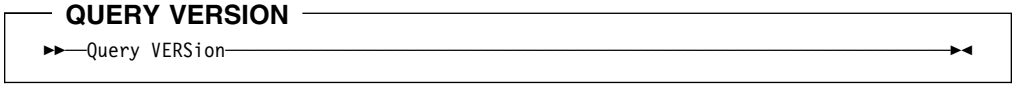
Response: TITLEn last set by *source: text*

where *source* is one of **profile, subcommand,** or **default**, and *text* is the text of the title.

QUERY VERSION

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand asks for the version level of the SVAA software.



Abbreviation: **Q VERS**

Response: VERSION last set by *source*: Shared Virtual Array Administrator *v.r.m*.

where *source* is one of **profile, subcommand,** or **default**, *v.r.m* is the version, release, and level of SVAA installed at your location.

REPORT PERFORMANCE Subcommand

* ***Subcommand of SIBBATCH or SIBADMIN***

This subcommand defines the parameters that control performance reporting. The abbreviation is REPORT PERF. You can select one or more reports. See Chapter 9, "Understanding the SVAA Reports" for the layout of performance reports.

Notes:

1. Performance data is not collected for Power PPRC Bridge devices.
2. PAV Alias devices are excluded from all data collection; thus, performance data is not reported for Aliases.

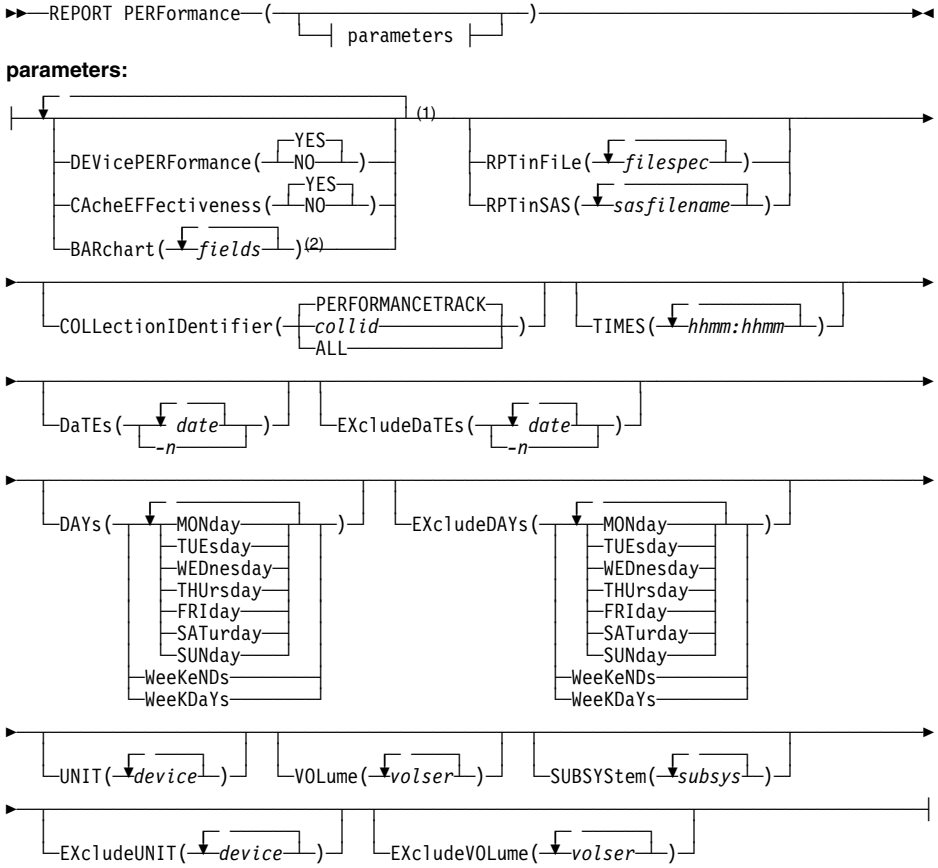
For Device Performance and Cache Effectiveness reports you can specify:

- Whether input for the report is in a SAS file or in an SMF- or user-managed file (RPTINFILE or RPTINSAS file parameter).
- The collection run from which data is to be reported (COLLECTIONIDENTIFIER parameter).
- The devices, volumes, and subsystems for which data is to be reported (UNIT, VOLUME, and SUBSYSTEM parameters).
- The devices, volumes, and subsystems for which data is not to be reported (EXCLUDEUNIT, EXCLUDEVOLUME, and EXCLUDESUBSYSTEM parameters).
- Whether data is to be reported for non-SVA subsystems (NONICEBERG parameter) or not (EXCLUDENONICEBERG parameter).
- The ranges of times of day for which data is to be reported (TIMES parameter).
- The individual dates or ranges of dates for which data is to be reported (DATES parameter).
- The individual dates or ranges of dates for which data excluded from reporting (EXCLUDEDATES parameter).
- The days of the week for which data is to be reported (DAYS parameter).
- The days of the week for which data is to be excluded from reporting (EXCLUDEDAYS parameter).
- Whether the report is to include device-performance data (DEVICEPERFORMANCE parameter).
- Whether the report is to include cache-effectiveness data (CACHEEFFECTIVENESS parameter).
- Whether the report is to include details for each functional device (DEVICEDETAIL parameter).
- Whether the reports are to be printed or produced in SAS form (SUPPRESSPRINT parameter).
- The SMF record type for your collection records (SMFNUMBER parameter).
- Whether an overall summary report is to be produced for the input data (OVERALL parameter).
- Whether an interval report is to be produced for the input data (INTERVALREPORT parameter).

- Whether a time-ordered report is to be produced for the input data (TIME-ORDERED parameter).
- Whether an exception report is to be produced for the input data (EXCEPTION parameter).
- Whether a summary by day, week, or month report is to be produced for the input data (SUMMARYBYDAY, SUMMARYBYWEEK, or SUMMARYBYMONTH parameters).
- Whether a bar-chart report is to be produced for the input data (BARCHART parameter).
- The names of the SAS device-detail files and subsystem totals files to be created for each day, week, or month as input to SAS/GRAPH for the interval, time-ordered, and summary reports (SASDEVICEFILE and SASSUBSYSTEM subparameters), and how many generations of these files are to be kept (GENS subparameter).
- The values to be flagged in an exception report (THRESHOLD, EXCEPTIONFILE, and PARTITION subparameters).

Note: Before using the REPORT PERF subcommand, you must allocate a data set with a DD of SASPGM.

REPORT PERFORMANCE



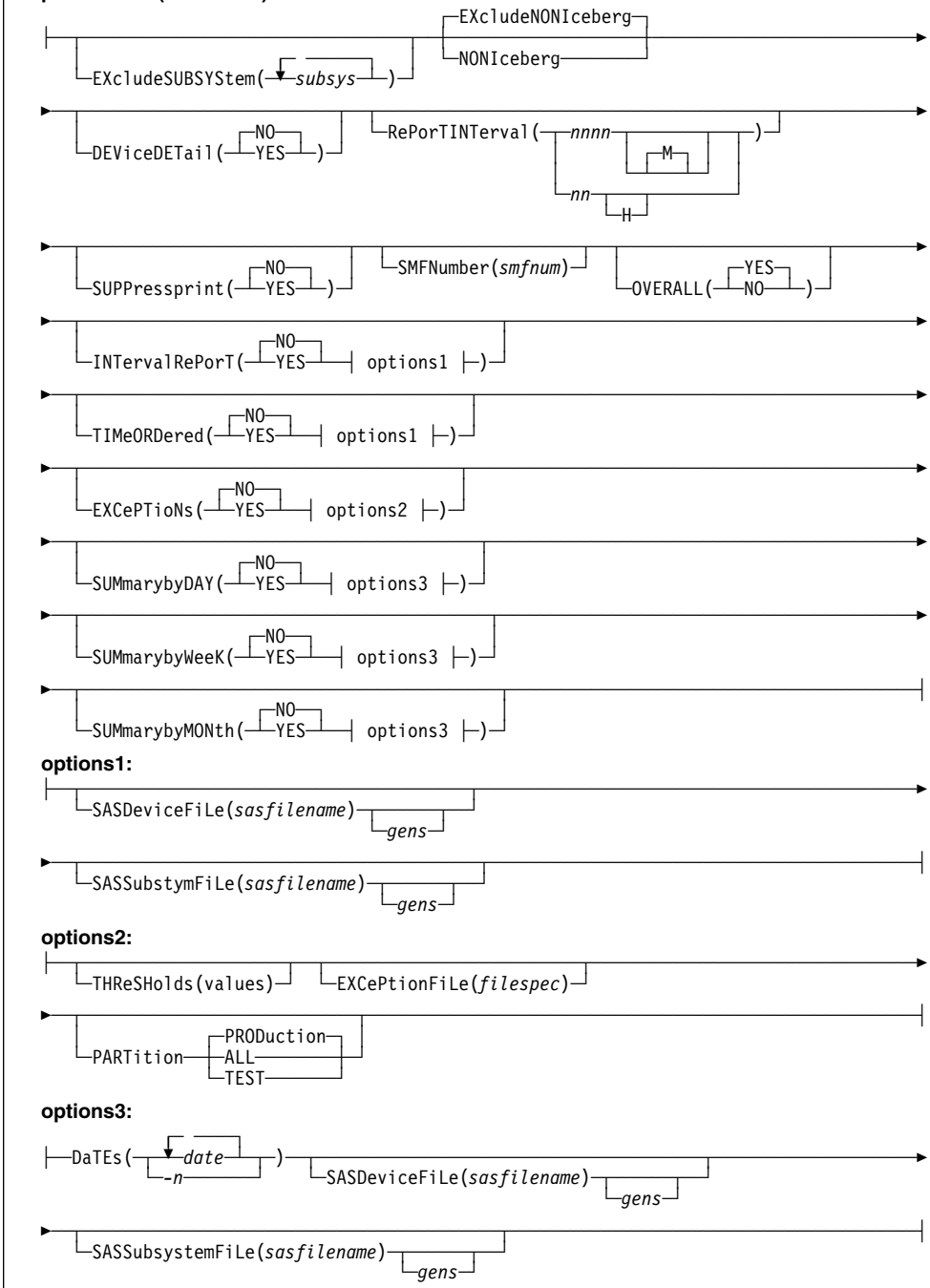
Notes:

1 You cannot specify DEVPERF(NO), CAEFF(NO), and also omit the BARCHART parameter. At least one type of report must be generated—by specifying YES for one of the first two parameters, by omitting at least one the first two parameters so that it defaults to YES, or by including the BARCHART parameter.

2 To obtain only a barchart report, specify the BARCHART parameter as well as specifying DEVPERF(NO), CAEFF(NO), and OVERALL(NO).

REPORT PERFORMANCE (Cont.)

parameters (continued):



Parameters and Subparameters

All the parameters and subparameters are optional. This section describes the parameters in alphabetical order. See page 8-58 for coding examples.

BARCHART (fields)

Optional. However, you must enter (or default) at least one of the BARCHART, DEVICEPERFORMANCE, or CACHEEFFECTIVENESS parameters.

This parameter requests reports in the form of histograms or bar charts. Refer to "Bar Charts" on page 9-12 for the bar-chart report layout. If you omit this parameter, you must use either the CAEFF or DEVPERF parameter.

Abbreviation: `BAR`

Values: *fields* specifies up to 19 names of fields for which bar charts are to be produced. Fields to be charted must be selected explicitly; there are no defaults.

You cannot use ranges or wildcard characters.

Refer to Appendix C, "Variables for Bar Chart Reports" for the field names that can be specified for producing bar charts.

Example: `BAR(IOHITPCT IOSVCTM IORTE)`

CACHEEFFECTIVENESS (YES | NO)

Optional. However, you must enter (or default) at least one of the BARCHART, DEVICEPERFORMANCE, or CACHEEFFECTIVENESS parameters.

This parameter specifies whether cache-effectiveness data is to be reported. Refer to Chapter 9, "Understanding the SVAA Reports" for report layouts. If you omit this parameter, you must use either the BAR or DEVPERF parameter.

Abbreviation: `CAEFF`

Values: `YES` specifies that cache-effectiveness data is to be reported

`NO` specifies that cache-effectiveness data is not to be reported

Default value: `YES`

Notes:

1. If you are specifying reporting for a non-SVA subsystem, you cannot specify this parameter as CAEFF(NO), because that is the only data reported for non-SVA subsystems.
2. If you specify CAEFF(YES) but there are no cache-effectiveness data in the files, Reporter prints the report header and an error message.

COLLECTIONIDENTIFIER (PERFORMANCETRACK | collid | ALL)

Required for OR data, optional for PT data

This parameter specifies the name assigned to the collection run on which this report is to be based. To produce a report for a specific OR collection run, use the collection identifier assigned to that collection. If you want the report to include statistics from all runs in the input file, specify the ALL value. Note, however, that if data was collected for the same devices during the same time period in more than one run, the resulting reports will be incorrect.

Abbreviation: `COLLID`

Values: *collid* specifies the unique name for the collection run to be reported.

Note: If you use lowercase characters and SET CASE(UPPER) is in effect, you must enclose the identifier in matching quotes.

ALL produces a report including the statistics from all runs in the input file.

PERFORMANCETRACK is a reserved identifier that specifies that the report is to be of PT data.

Default value: **PERFORMANCETRACK**

Example **COLLID(ORCOLLID)** specifies that the collection identifier for this OR collection run is **ORCOLLID**.

DATES (dates | -n)

Optional

This parameter or subparameter specifies start-stop date ranges for reporting to cover. (You can specify the dates for all reports with the DATES parameter or the dates for specific summary reports with the DATES subparameter of the SUMDAY, SUMWK, or SUMMON parameters.) Use the DAYS parameter to control the days of the week to be reported on, over the date ranges that you specify here.

Abbreviation: **DTE**

Values: *dates* specifies up to 32 single dates or pairs of starting and ending dates for data reporting to cover, each date being in the form *ddmmmyyyy*. You can specify dates in the future.

-n specifies the date to start reporting as a relative displacement in days before the current date, where *n* is the number of days. You can specify the current date as either 0 or -0. You can specify a range of relative displacements.

Default value: The dates or date ranges you specify with the DATES parameter of REPORT PERF apply to all reports. However, the dates or date ranges you specify with the DATES subparameter of the SUMDAY, SUMWK or SUMMON parameter of REPORT PERF or on the SPECIFY SUMMARY REPORT OPTIONS panel apply only to those reports. If you specify neither the DATES subparameter of a summary report parameter nor the DATES parameter, Reporter produces reports for all data in the history file.

Examples: **DTE(26JUN1999:15JUL1999)**

DATES(0) specifies today

DTE(-1) specifies yesterday

DTE(-6:0) specifies the week ending with today

DTE(02May1999:-3) specifies the dates from May 2, 1999 until 3 days before the current date.

Notes:

1. The *-n* value always refers to a number of days, even when you use DTE as a subparameter of the SUMDAY, SUMWK or SUMMON parameters.
2. Entering single dates or a range of dates when DTE is a subparameter of the SUMDAY, SUMWK or SUMMON parameters specifies that the weekly or monthly report is to summarize statistics only for those days.
3. For more information about specifying dates, see page 6-8. In particular, see Note 11 on page 6-9. (See also Example 4 on page 8-59.)

DAYS (days | WeeKeNDs | WeeKDaYs)

Optional. If you enter this parameter, you cannot enter the EXDAY parameter.

This parameter specifies the days of the week that the reports are to cover. The days of the week you specify control the days for reporting within the date ranges specified by the DATES parameter.

Abbreviation: **DAY**

Values: *days* specifies any individual days of the week, spelled out (**MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY, SUNDAY**) or the first three letters of any day of the week (**MON, TUE, WED, THU, FRI, SAT, SUN**)

WEEKENDS (or **WKND**) specifies Saturday and Sunday.

WEEKDAYS (or **WKDY**) specifies Monday through Friday.

Note: You can enter up to 7 specific days of the week for this parameter, or **WKND**, or **WKDY**, but no combination of these three options.

Default value: All days in the history files or the raw data files.

Example: **DAY(MON WED FRI)**

DEVICEDETAIL (YES | NO)

Optional

This parameter specifies whether to print, for all selected reports, detail lines for the individual functional devices.

Abbreviation: **DEVDET**

Values: **YES** requests report details for individual devices

NO requests partition and subsystem totals only

Default value: **NO**

Note: Even if you specify **DEVDET (NO)** for exception reports, Reporter ignores your specification and provides details.

DEVICEPERFORMANCE (YES | NO)

Optional. However, you must enter (or default) at least one of the BARCHART, DEVICEPERFORMANCE, or CACHEEFFECTIVENESS parameters.

This parameter specifies whether device-performance data is to be reported. If you omit this parameter, you must use either the CAEFF or BAR parameter.

Abbreviation: **DEVPERF**

Values: **YES** specifies that performance data is to be reported.

NO specifies that performance data is not to be reported.

Default value: **YES**

Notes:

1. Reporter never reports device-performance data for non-SVA devices.
2. If you specify **DEVPERF(YES)** but there are no device-performance data in the files, Reporter puts out the report header and an error message.

EXCEPTIONFILE (filespec)

Optional

This subparameter of the EXCPTN parameter specifies the filespec of an optional file containing the values for exception thresholds. Any exception parameter values explicitly included as part of the THRSH parameter override the values for the same exception parameter found in the exception file.

Abbreviation: **EXCPFL**

Values: *filespec* specifies the file that contains the threshold values.

You can use the *style* prefix. If you specify it, it must be DDN. The DD name identifies a DD command in the SIBBATCH step of the report JCL. (See "Using the Style Prefix" on page 8-8 for more information about the style prefix.)

Default value: If you omit this subparameter, Reporter assumes that there is no exceptions file.

Note: See "Specifying Exception Thresholds" on page 6-10 for information about specifying thresholds, the values you can specify, and the default values.

EXCEPTIONS (NO | YES [subparms])

Optional

This parameter specifies whether you want Reporter to produce exception reports for SVA devices. If you specify YES for this parameter, either specify in the file named by the EXCPFL parameter the values you want to include in the exception report, or specify the values individually with the THRSH parameter.

Abbreviation: **EXCPTN**

Values: **NO** specifies that you do not want an exception report.

If you specify **NO** (or omit the **YES** or **NO** value) Reporter ignores the **THRSH, EXCPFL,** and **PART** subparameters if you supply them.

YES specifies that Reporter is to produce an exception report. If you specify YES, you can also specify the THRESH, EXCPFL, and PART subparameters.

subparms are THRSH, EXCPFL, or PART subparameters.

Default value: **NO**

EXCLUDEDATES (dates | -n)

Optional

This parameter lists the dates that are to be excluded from all the reports. The EXDTE parameter has precedence over the other date-selection parameters, including the DTE subparameter of the SUMDAY, SUMWK, and SUMMON parameters.

Abbreviation: **EXDTE**

Values: *dates* specifies up to 32 dates or ranges of starting and ending dates for which data reporting to be excluded, each date being in the form *ddmmmyyyy*.

-n specifies an exclusion date as a relative displacement in days before the current date, where *n* is the number of days. You can specify the current date as either 0 or -0. You can specify a range of relative exclusion dates.

Example:
```
EXDTE(15JAN1999)
EXDTE(28FEB1999 1MAR1999 -5:-1)
```

Notes:

1. The *-n* value always refers to a number of days, even when you use DTE as a subparameter of the SUMDAY, SUMWK or SUMMON parameters.
2. For more information about specifying dates, see page 6-8.

EXCLUDEDAYS (days | WeeKeNDs | WeeKDaYs)

Optional. If you enter this parameter, you cannot enter the DAY parameter.

This parameter lists the days of the week that are to be excluded from all the reports. The days of the week you specify with the DAYS and EXDAY parameters control the days to be reported on, over the date ranges you specify with the DTE and EXDTE parameters.

Abbreviation: **EXDAY**

Values: *days* specifies any individual days of the week, spelled out (**MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY, SUNDAY**) or the first three letters of any day of the week (**MON, TUE, WED, THU, FRI, SAT, SUN**)

WEEKENDS (or **WKND**) specifies Saturday and Sunday.

WEEKDAYS (or **WKDY**) specifies Monday through Friday.

Note: You can enter up to 7 specific days of the week for this parameter, or **WKND**, or **WKDY**, but no combination of these three options.

Example:
```
EXDAY (MON SUN)
```

EXCLUDENONICEBERG

Optional. If you enter this parameter, you cannot enter the NONIB parameter.

This parameter specifies that Reporter is not to report data from non-SVA disk subsystems. This parameter is the default; Reporter only reports data for non-SVA subsystems if you specify the NONIB parameter, or if the UNIT or VOLUME parameter identifies a non-SVA device.

Abbreviation: **EXNONIB**

Values: None

Default value: **EXNONIB**

EXCLUDESUBSYSTEM (subsys)

Optional

This parameter specifies the subsystems to be excluded from data reporting.

Abbreviation: **EXSUBSYS**

Values: *subsys* specifies the names of up to 32 installed SVA subsystems to be excluded. Use exactly the same uppercase or lowercase characters as defined for the subsystem name.

You can use wildcard characters and ranges.

If a subsystem name contains lowercase characters and SET CASE(UPPER) is in effect, you must enclose the name in matching quotes.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter reports data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

EXCLUDEUNIT (device)

Optional

This parameter specifies devices (as they are defined to the host operating system) to be excluded from data reporting. You can specify up to 4096 devices on the V2X or up to 1024 on earlier SVAs. Devices to be excluded modify and override the list of devices to be included.

Abbreviation: **EXUNIT**

Values: *device* specifies host device numbers (1 to 4 hexadecimal digits per device). You can pad to the left with zeros. Reporter treats lowercase the same as uppercase.

You can use wildcard characters and ranges.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter reports data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

EXCLUDEVOLUME (volser)

Optional

This parameter specifies the volume serial numbers of devices to be excluded from data reporting. You can specify up to 4096 volsers on the V2X or up to 1024 on earlier SVAs. Volumes to be excluded modify and override the list of volumes to be included.

Abbreviation: **EXVOL**

Values: *volser* specifies volume serial numbers of devices to be excluded. SVAA converts all alphabetics in volsers to uppercase.

You can use wildcard characters and ranges.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter reports data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

INTERVALREPORT (NO | YES [subparms])

Optional

This parameter specifies that interval performance reports are to be produced.

Abbreviation: **INTRPT**

Values: **NO** specifies that interval reports are not to be produced.

If you specify **NO** (or omit the **YES** or **NO** value) Reporter ignores the SASDFL and SASSFL subparameters if you supply them.

YES specifies that interval performance reports are to be produced. If you specify **YES** you can also specify the SASDFL and SASSFL subparameters.

subparms are SASDFL or SASSFL subparameters.

Default value: **NO**

NONICEBERG

Optional. If you enter this parameter, you cannot enter the EXNONIB parameter.

This parameter specifies that Reporter is to include data for non-SVA disk subsystems, except for devices excluded by EXUNIT or EXVOL. Only cache statistics available from the 3990 Read Subsystem Data command are produced. There is no non-SVA device selection provided at the control unit or SSID level.

Abbreviation: **NONIB**

Values: None

Default value: **EXNONIB**

Note: If you enter this parameter, you must also enter CAEFF(YES) or BARCHART(*fields*).

OVERALL (YES | NO)

Optional

This parameter specifies whether the overall summary report is to be produced.

Abbreviation: None

Values: **YES** requests Reporter to produce an overall summary report.

NO specifies that there is to be no overall summary report.

Default value: **YES**

Note: You cannot specify all of OVERALL(NO), DEVPERF(NO), and CAEFF(NO) unless you specify the BARCHART parameter.

PARTITION (ALL | PRODUCTION | TEST)

Optional

This subparameter of the EXCPTN parameter specifies the partitions for which exception reports are to be produced.

Abbreviation: **PART**

Values: **ALL** specifies reports for both the Test and Production partitions.

TEST specifies that statistics for only the Test partition are to be reported.

PROD specifies that statistics for only the Production partition are to be reported.

Default value: **PROD**

REPORTINTERVAL (nnnn[M] | nn[H])

Optional

This parameter specifies the data reporting interval. The parameter applies only to Interval, Time-Ordered, Performance Bar Chart, and Exception reports.

Abbreviation: **RPTINT**

Values: *nnnn*[**M**] specifies the number of minutes (1 to 1440) in the reporting interval. The numeric portion of the value can be up to four digits; leading zeros are allowed.

nn[**H**] specifies number of hours (1 to 24) in the reporting interval. The numeric portion of the value can be up to two digits; one leading zero is allowed.

The value of *nnnn* must be a multiple of the collection interval used in collecting the input data, as specified with the INT parameter of the COLL ORD or COLL PT subcommands. *nnnn* must also be a multiple of the summarization interval, as specified with the INT parameter of the SUMM HIST subcommand. If *nnnn* is not such a multiple, unexpected results may occur.

Default value: 120**M**

RPTINFILE (filespec)

Optional. If you enter this parameter, you cannot enter the RPTINSAS parameter.

This parameter specifies that input for the report is in one or more files in SMF format.

Abbreviation: **RPTFL**

Values: *filespec* identifies the files containing the input. There is no maximum number of input files other than the restrictions on command lengths and REXX line lengths.

If you use a style prefix, you can use only the DDN prefix (see "Using the Style Prefix" on page 8-8 for more information).

Default value: If you omit both the RPTFL and the RPTSAS parameters, Reporter assumes that the report input is in the **DDN:RPTIN** file.

Note: The *style* prefix is optional. The DD name identifies a DD statement in the SAS step of the report JCL (see the **RPTIN** statement of the sample JCL in "Requesting Performance Reports" on page A-4.) The value you use must agree with the value in the JCL. You can specify only one *ddname* (that is, you must concatenate all input files in one DD statement).

RPTINSAS (sasfilename)

Optional. If you enter this parameter, you cannot enter the RPTINFILE parameter.

This parameter specifies that input for the report is in the SAS performance history file (and not in a user-managed or SMF file), and gives the SAS name of the input file.

Abbreviation: **RPTSAS**

Values: Each *sasfilename* identifies a set of four files containing the input. *sasfilename* cannot exceed 7 characters because Reporter adds one character to differentiate the four types of SAS input file (see "Summarizing Logged Data" on page 5-4).

There is no maximum number of input files other than the restrictions on command lengths and REXX line lengths. (For more information about SAS file names, see "Specifying SAS Data Sets" on page 8-9.)

Default value: If you omit both the RPTFL and the RPTSAS parameters, Reporter assumes that the report input is in the **DDN:RPTIN** file.

SASDEVICEFILE (sasfilename) [gens]

Optional

This subparameter of the INTRPT, SUMDAY, SUMWK, SUMMON or TIMORD parameter specifies the name for the SAS device-detail file to be created as input to SAS/GRAPH. If you request both device performance and cache effectiveness reports, the reports are merged into a single SAS file.

Abbreviation: **SASDFL**

Values: *sasfilename* [*gens*] specifies the name of the SAS device file. See "Specifying SAS Data Sets" on page 8-9 for a discussion of specifying SAS files.

The length of *sasfilename* plus the number of digits in *gens* cannot exceed 8 characters.

gens specifies the number generations of SAS files to keep, specified as 1 to 3 decimal digits (1-999). See Note 2 on page 8-9.

Default value: The default number of generations is 5.

SASSUBSYSTEMFILE (sasfilename) [gens]

Optional

This subparameter of the INTRPT, SUMDAY, SUMWK, SUMMON or TIMORD parameter specifies the name for the SAS subsystem totals file to be created as input to SAS/GRAPH.

Abbreviation: **SASSFL**

Values: *sasfilename* [*gens*] specifies the file to contain the SAS subsystem totals. See "Specifying SAS Data Sets" on page 8-9 for a discussion of how SAS files can be specified.

The length *sasfilename* plus the number of digits in *gens* cannot exceed 8 characters.

gens specifies the number of generations of SAS files to keep, specified as one to three decimal digits (1-999). See Note 2 on page 8-9.

Default value: The default number of generations is 5 files.

SMFNUMBER (smfnum)

Conditional

This parameter, which is usually not needed, specifies the SMF record type assigned by your system programmer for your data collection records in SMF.

The SMF record type is normally specified to Reporter using the SUBPARM parameter in the SMFPRMxx member of SYS1.PARMLIB. This parameter is usually optional, but must be specified if the input file contains SMF records and if one or more of the following is true:

- The SMF data was created on another host with a different SVAA SMF number (as when you are producing the report on a VM system but the data was collected on an OS/390 system).
- The OS/390 host on which the data is being processed does not contain an entry for SVAA in the SIBPRMxx member.
- The OS/390 host on which the data is being processed does not have an initialized and active SVAA subsystem.

Abbreviation: **SMFN**

Values: *smfnum* specifies SMF record type as a whole number ranging from 128 to 255 inclusive.

Default value: None.

SUBSYSTEM (subsys)

Optional

This parameter specifies one or more SVA subsystem names.

Abbreviation: **SUBSYS**

Values: *subsys* specifies the names of up to 32 installed SVA subsystems. Use exactly the same uppercase and lowercase characters as defined for the subsystem name.

You can use wildcard characters and ranges.

If a subsystem name contains lowercase characters and SET CASE(UPPER) is in effect, you must enclose the name in matching quotes.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter reports data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

SUMMARYBYDAY (NO | YES [subparms])

Optional

This parameter specifies whether you want daily summary reports.

Abbreviation: **SUMDAY**

Values: **NO** specifies that you do not want a daily summary report. Reporter ignores the DTE, SASDFL and SASSFL subparameters if you supply them.

YES specifies that you want a summary report for each day. If you specify **YES**, you can also specify the DTE, SASDFL and SASSFL subparameters.

subparms are DTE, SASDFL, or SASSFL subparameters.

Default value: **NO**

Example: **SUMDAY(YES DTE(-3))** on a subcommand issued on a Tuesday requests a report that will contain four lines, one for each of the preceding Saturday, Sunday, and Monday, and one for Tuesday.

SUMMARYBYMONTH (NO | YES [subparms])

Optional

This parameter specifies whether you want monthly summary reports. A monthly summary report has an entry summarizing the data for each calendar month in the period specified by the DTE subparameter. (Therefore, consecutive lines of the report may summarize the data for periods ranging from 28 to 31 days.)

Abbreviation: **SUMMON**

Values: **NO** specifies that you do not want a monthly summary report. If you specify **NO** (or omit the **YES** or **NO** value) Reporter ignores the DTE, SASDFL and SASSFL subparameters if you supply them.

YES specifies that you want a summary report for each month. If you specify **YES**, you can also specify the DTE, SASDFL and SASSFL subparameters.

subparms are DTE, SASDFL, or SASSFL subparameters.

Default value: **NO**

Example: **SUMMON(YES DTE(-10))** on a subcommand issued on April 5, 1999 requests a report that will contain the following entries:

- A line summarizing the data for April 1 through April 5, 1999
- A line summarizing the data for March 27 through March 31, 1999 (the previous month).

SUMMARYBYWEEK (NO | YES [subparms])

Optional

This parameter specifies whether you want weekly summary reports. A weekly summary report has an entry summarizing the statistics for each week (Sunday through Saturday) in the period specified by the DTE subparameter.

Abbreviation: **SUMWK**

Values: **NO** specifies that you do not want a daily summary report. If you specify **NO** (or omit the **YES** or **NO** value) Reporter ignores the DTE, SASDFL and SASSFL subparameters if you supply them.

YES specifies that you want a summary report for each day. If you specify **YES**, you can also specify the DTE, SASDFL and SASSFL subparameters.

subparms are DTE, SASDFL, or SASSFL subparameters.

Default value: **NO**

Example: **SUMWK(YES DTE(-10))** on a subcommand issued on a Wednesday requests a report that will contain the following entries:

- An entry summarizing the data for Sunday through Wednesday of the current week
- An entry summarizing the data for Monday through Saturday of the previous week.

Note: Using the DTE subparameter with a single date specifies that the report is to contain data only for that date.

SUPPRESSPRINT (YES | NO)

Optional

This parameter specifies whether you want to suppress printed output of the reports, leaving only SAS output for input to the SAS/GRAPH program.

Abbreviation: **SUPP**

Values: **YES** suppresses printed output from the Reporter report. Reporter produces only SAS output (that is, output in SAS data set form), as specified by the SASDFL or SASSFL parameters).

If you specify **YES**, you must also specify either SASDFL or SASSFL.

Note: If you do not request SAS output by specifying the SASDFL or SASSFL parameter, Reporter ignores SUPP(YES).

NO allows printed output from Reporter.

Default value: **NO**

THRESHOLDS (values)

Optional

This subparameter of the EXCPTN parameter allows you to select individual fields for reporting for values that exceed, lie between, lie outside, or fall below certain specified thresholds. If you specify individual exception thresholds, Reporter places asterisks beside those values in the reports that fall outside the threshold values. If you specify EXCEPTION(YES) but omit both this parameter and the EXCEPTIONFILE parameter, Reporter monitors device performance and cache effectiveness, using the default values for all thresholds.

See "Specifying Exception Thresholds" on page 6-10 for more information. See Appendix B, "Exception Thresholds" for the names and descriptions of fields available for exception reporting, and for their default thresholds.

Abbreviation: **THRSH**

Values: See Appendix B, "Exception Thresholds."

TIMEORDERED (NO | YES [subparms])

Optional

This parameter specifies whether you want time-ordered reports.

Abbreviation: **TIMORD**

Values: **NO** specifies that Reporter is not to produce time-ordered reports.

If you specify **NO** (or omit the **YES** or **NO** value) Reporter ignores the SASDFL and SASSFL subparameters if you supply them.

YES specifies that Reporter is to produce time-ordered reports. If you specify **YES**, you can also specify the SASDFL or SASSFL subparameters.

subparms are SASDFL or SASSFL subparameters.

Default value: **NO**

TIMES (hhmm:hhmm)

Optional

This parameter specifies the start-stop time-range pairs for data reporting. Reporter produces a separate report for each time range (or part of a time range) for which data exists, for the dates specified by the DATES parameter (note that SVAA does *not* treat dates and times as starting and ending date-time pairs).

Values: *hhmm:hhmm* specifies up to 16 pairs of times for which data collection is to be started and stopped (in hours and minutes, using the 24-hour clock). The first value in the pair is the starting time and must be between 0000 and 2359; the second is the stopping time and must be between 0001 and 2400. Separate the two times by a colon; separate multiple ranges by spaces. If the stopping time is earlier than the starting time, reporting spans midnight. You cannot specify overlapping time-range pairs. You can specify ranges in any order.

Default value: If you do not specify the TIMES or DUR parameters, Reporter reports all data in the history file. Specifying 0000:2400 also results in all data being reported.

Note: See "Choosing Time Ranges for Reporting" on page 6-6 for more examples and information.

UNIT (device)

Optional

This parameter specifies devices (as they are defined to the host operating system) for which data is to be reported. You can specify up to 4096 devices on the V2X or up to 1024 on earlier SVAs.

Values: *device* specifies host device numbers (1 to 4 hexadecimal digits per device). You can pad to the left with zeros. Reporter treats lowercase the same as uppercase.

You can use wildcard characters and ranges.

Examples: **UNIT(2D4 2D5 3A0:3BF)** requests data reporting for devices **2D4, 2D5** and all devices with addresses in the range **3A0** to **3BF**.

UNIT(3*) requests data reporting for all devices with the first digit of the address equal to 3.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter reports data for all devices on all SVA subsystems.
2. See "Specifying Devices for Data Collection" on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

VOLUME (volser)

Optional

This parameter specifies the volume serial numbers of devices for which data is to be reported. You can specify up to 4096 volsers on the V2X or up to 1024 on earlier SVAs.

Abbreviation: **VOL**

Values: *volser* specifies volume serial numbers of devices to be included. SVAA converts all alphabetics in volsers to uppercase.

You can use wildcard characters and ranges.

Example: **VOL(PROD01:PROD25)** requests data reporting for all devices with volume serial number **PROD01** through **PROD25**.

VOL(PR*) requests data reporting for all devices with volume serial numbers starting with **PR**, while **VOL(PR?D*)** requests inclusion of all devices with volume serial numbers starting with **PR** and with a fourth byte of **D**.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter reports data for all devices on all SVA subsystems.
2. See "Specifying Devices for Data Collection" on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

Examples of REPORT PERF

Example 1. Performance reports using all defaults

```
REPORT PERF
```

This example requests performance reporting with all defaults. The RPTIN DD statement in the JCL points to the input data. All dates and times present in the input file are included in the output report. Reporter produces an overall summary report containing partition and subsystem totals only for all subsystems (no device details). Reporter reports data from the PT data-collection run. This can be done in SIBBATCH only.

Example 2. Performance reports for a date range with exception thresholds

```
REPORT PERF-
    (RPTFL(DDN:PERFILE)-
    DTE(-10:0)-
    EXDAY(WKND)-
    EXDTE(15FEB1999)-
    TIMES(1200:1500)-
    SUBSYS(SYSPROD*)-
    EXVOL(TSO001:TSO005)-
    DEVDET(YES)-
    RPTINT(1H)-
    TIMORD(YES)-
    EXCPTN(YES-
       THRSH(-
          (IOSVCTM GT 15)-
          (IOHITPCT LT *) )-
       PART(ALL) ))
```

This example requests performance reporting. The DD name of the input data file is PERFILE. The report period is the last 11 elapsed days including today, but excluding weekends and Feb. 15, 1999. The period from 12 noon to 3 p.m. is included for each day. Statistics are reported for all subsystems with names starting with **SYSPROD**; however, volumes **TSO001** through **TSO005** are excluded. Device-detail statistics are to be produced for all reports. The reporting interval is one hour. A time-ordered report, an overall summary report, and an exception report are to be produced. The following thresholds are monitored for both Test and Production partitions:

- I/O service time greater than 15 ms
- Cache I/O hit percentage less than the default threshold (70)

The data from the PT data-collection run is reported.

Example 3. Performance reports with SMF input and excluded dates

```
REPORT PERF-
    (RPTFL(DDN:SMFDAILY)-
    DTE(01MAY1998:30OCT1998)-
    DAY(WKDY)-
    EXCDTE(29MAY1998-
            03JUL1998-
            04JUL1998-
            04SEP1998)-
    TIMES(0800:1600-
          1600:2400)-
    EXSUBSYS(SUBSYS09)-
    RPTINT(2H)-
    INTRPT(YES)-
    OVERALL(NO)-
    SUMDAY(YES)-
    SMF(220) )
```

This example requests performance reporting. The DD name of the input data file is SMFDAILY. (This example illustrates that SMF data can be used as input to the report program.) The SVAA SMF record number at this installation is 220.

The reporting period encompasses six months from May 1, 1998 to October 30, 1998. Only weekdays are included, and holidays are excluded, as is July 3, 1998. Note that this entire period need not be present in the input file; the DTE parameter sets limits, not requirements. There are separate reports produced for first shift (8 a.m. to 4 p.m.) and for second shift (4 p.m. to midnight). Reports are produced for all subsystems except **SUBSYS09**.

An interval report and daily summary report are requested. The reporting interval for the interval report will be two hours. An overall summary report is not to be produced. All reports will give partition and subsystem totals only (no device details). The data from the PT data-collection run is reported.

Example 4. Performance reports based on data spanning midnight

```
REPORT PERF-
    (DTE(01SEP1998))
```

Assume that the data for the requested report was collected by specifying:

```
COLLECT PTDATA ( INTERVAL(1H) SYNCHRONIZE(30))
```

The first collection interval involved in the report ends at 12:30 a.m., but has a timestamp of 2330 on the previous day. Therefore the report includes data for both August 31, 1998 and September 1, 1998.

Example 5. Performance reports from summarized history file

```
REPORT PERF-
    (RPTFL(DDN:SUMMFILE )-
    DTE(-91:0)-
    DAY(WKDY)-
    EXDTE(01JAN1999-
            15JAN1999-
            19FEB1999-
            15MAR1999-
            16MAR1999)-
    TIMES(1600:2400)-
    SUMDAY(YES-
        DTE(-14:0) )-
    SUMWK(YES)-
    SUMMON(YES) )
```

This example requests performance reporting. The input data file is the current version of the summarized history file, with a DD name SUMMFILE. Second shift (4 p.m. to midnight) is reported for all weekdays, excluding holidays. Statistics are reported for all subsystems. Partition and subsystem totals only are produced (no device details). Both Device Performance and Cache Effectiveness statistics are reported.

A summary by day report is produced for the last 15 days; a weekly summary report is produced for the last 12 weeks; and a monthly summary report is produced for the last 3 months present in the input file. The data from the PT data-collection run is reported.

REPORT SPACEUTILIZATION Subcommand

**** Subcommand of SIBBATCH or SIBADMIN***

This subcommand defines the parameters for Space Utilization reporting. The abbreviation is REPORT SPACEU.

See Chapter 9, “Understanding the SVAA Reports” for the layout of the Space Utilization report.

Notes:

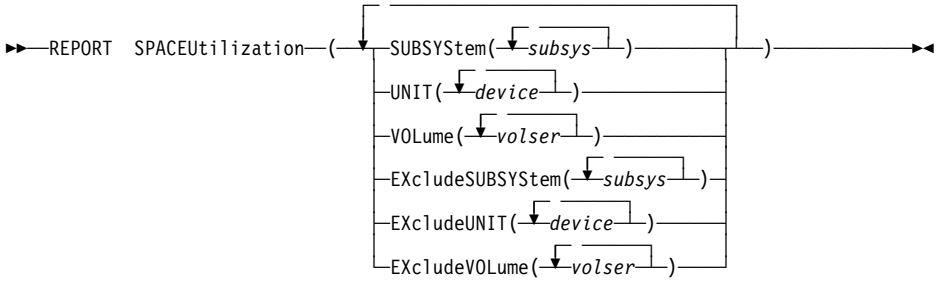
1. Although Reporter can report on space utilization for Power PPRC Bridge devices, it may not make sense to do so. Bridge devices must not be used for user data, so there is little or no value in generating reports for these devices.
2. PAV Alias devices are excluded from all data collection; thus, space utilization data is not reported for Aliases.

For the Space Utilization reports, you can specify:

- The functional devices, volumes, and subsystems for which data is to be reported (UNIT, VOLUME, and SUBSYSTEM parameters).
- The functional devices, volumes, and subsystems for which data is not to be reported (EXCLUDEUNIT, EXCLUDEVOLUME, and EXCLUDESUBSYSTEM parameters).

Note: Before using the REPORT SPACEU subcommand, you must allocate a data set with a DD of SASPGM.

REPORT SPACEUTILIZATION



Parameters

All of the parameters are optional. See page 8-64 for examples of coding the parameters.

EXCLUDESUBSYSTEM (subsys)

Optional

This parameter specifies the subsystems to be excluded from data reporting.

Abbreviation: **EXSUBSYS**

Values: *subsys* specifies the names of up to 32 installed SVA subsystems to be excluded. Use exactly the same uppercase or lowercase characters as defined for the subsystem name.

You can use wildcard characters and ranges.

If a subsystem name contains lowercase characters and SET CASE(UPPER) is in effect, you must enclose the name in matching quotes.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter reports data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

EXCLUDEUNIT (device)

Optional

This parameter specifies devices (as they are defined to the host operating system) to be excluded from data reporting. You can specify up to 4096 devices on the V2X or up to 1024 on earlier SVAs. Devices to be excluded modify and override the list of devices to be included.

Abbreviation: `EXUNIT`

Values: *device* specifies host device numbers (1 to 4 hexadecimal digits per device). You can pad to the left with zeros. Reporter treats lowercase the same as uppercase.

You can use wildcard characters and ranges.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter reports data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

EXCLUDEVOLUME (volser)

Optional

This parameter specifies the volume serial numbers of devices to be excluded from data reporting. You can specify up to 4096 volsers on the V2X or up to 1024 on earlier SVAs. Volumes to be excluded modify and override the list of volumes to be included.

Abbreviation: `EXVOL`

Values: *volser* specifies volume serial numbers of devices to be excluded. SVAA converts all alphabetics in volsers to uppercase.

You can use wildcard characters and ranges.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter reports data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

SUBSYSTEM (subsys)

Optional

This parameter specifies one or more SVA subsystem names.

Abbreviation: **SUBSYS**

Values: *subsys* specifies the names of up to 32 installed SVA subsystems. Use exactly the same uppercase and lowercase characters as defined for the subsystem name.

You can use wildcard characters and ranges.

If a subsystem name contains lowercase characters and SET CASE(UPPER) is in effect, you must enclose the name in matching quotes.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter reports data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

UNIT (device)

Optional

This parameter specifies devices (as they are defined to the host operating system) for which data is to be reported. You can specify up to 4096 devices on the V2X or up to 1024 on earlier SVAs.

Values: *device* specifies host device numbers (1 to 4 hexadecimal digits per device). You can pad to the left with zeros. Reporter treats lowercase the same as uppercase.

You can use wildcard characters and ranges.

Examples: **UNIT(2D4 2D5 3A0:3BF)** requests data reporting for devices **2D4, 2D5** and all devices with addresses in the range **3A0** to **3BF**.

UNIT(3*) requests data reporting for all devices with the first digit of the address equal to 3.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter reports data for all devices on all SVA subsystems.
2. See “Specifying Devices for Data Collection” on page 4-11 for examples and interactions between device inclusion and exclusion parameters.
3. You cannot specify non-SVA 3380 or 3390 devices.

VOLUME (volser)

Optional

This parameter specifies the volume serial numbers of devices for which data is to be reported. You can specify up to 4096 volsers on the V2X or up to 1024 on earlier SVAs.

Abbreviation: **VOL**

Values: *volser* specifies volume serial numbers of devices to be included. SVAA converts all alphabetics in volsers to uppercase.

You can use wildcard characters and ranges.

Example: **VOL(PROD01:PROD25)** requests data reporting for all devices with volume serial number **PROD01** through **PROD25**.

VOL(PR*) requests data reporting for all devices with volume serial numbers starting with **PR**, while **VOL(PR?D*)** requests inclusion of all devices with volume serial numbers starting with **PR** and with a fourth byte of **D**.

Notes:

1. If you don't specify a selection parameter (UNIT, VOL, SUBSYS, EXUNIT, EXVOL, EXSUBSYS), Reporter reports data for all devices on all SVA subsystems.
2. See "Specifying Devices for Data Collection" on page 4-11 for examples and interactions between device inclusion and exclusion parameters.

Examples of REPORT SPACEU

Example 1. Space Utilization reporting with all defaults

```
REPORT SPACEU
```

This example requests Space Utilization reporting with all defaults. Space utilization statistics are printed for all devices attached to all SVA subsystems.

Example 2. Space Utilization reporting for specific functional devices

```
REPORT SPACEU-
    (EXUNIT(200:222) )
```

This example requests Space Utilization reporting for all functional devices attached to all SVA subsystems, excluding device addresses 200 through 222.

SEND Subcommands

The SEND subcommand sends command requests to tasks executing in the SVAA address space.

SEND DSR

* ***Subcommand of SIBBATCH or SIBADMIN***
* ***System Operator command***

The SEND DSR subcommand sends a command request to the DDSR Main Task.

SEND DSR

Values: *command* is any valid command (1 to 127 characters) enclosed in matching quotes (single or double).

Default value: None

Examples: From SIBBATCH or SIBADMIN:

```
SEND DSR('REL INTDAT("task123")')
```

From the OS/390 operator console:

```
F taskname,SEND DSR('STOP RELEASE(RELID(*))')
```

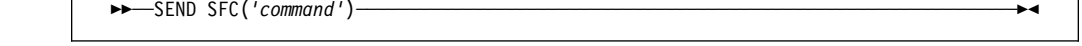
SEND SFC

* ***Subcommand of SIBBATCH or SIBADMIN***
* ***System Operator command***

The SEND SFC subcommand sends a command request to the Server Framework Component (SFC) task.

Note: Sending SFC SET subcommands to change parameters such as CONFIGPATH or SIBPORT has no impact; it does not change the Java address space. Java address space values are taken only from parmlib member SIBSFCxx.

SEND SFC



Values: *command* is any valid command (1 to 127 characters) enclosed in matching quotes (single or double).

Default value: None

Examples: From SIBBATCH or SIBADMIN:

```
SEND SFC('STOP SFC')
```

From the OS/390 operator console:

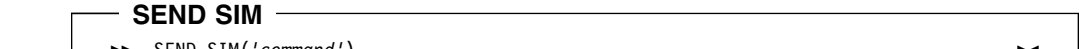
```
F taskname,SEND SFC('STOP SFC')
```

SEND SIM

* ***Subcommand of SIBBATCH or SIBADMIN***
* ***System Operator command***

The SEND SIM subcommand sends a command request to the SVAA Service Information Message (SIM) task.

SEND SIM



Values: *command* is any valid command (1 to 127 characters) enclosed in matching quotes (single or double).

Default value: None

Examples: From SIBBATCH or SIBADMIN:

```
SEND SIM("QUERY SUBSYSTEM")
```

From the OS/390 operator console:

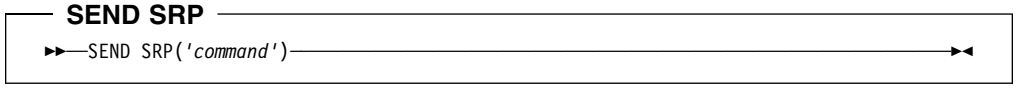
```
F taskname,SEND SIM("QUERY SUBSYSTEM")
```

SEND SRP

* ***Subcommand of SIBBATCH or SIBADMIN***
* ***System Operator command***

The SEND SRP subcommand sends a command request to the Reporter Data Collection task.

SEND SRP



Values: *command* is any valid command (1 to 127 characters) enclosed in matching quotes (single or double).

Default value: None

Examples: From SIBBATCH or SIBADMIN:

```
SEND SRP("D COLLTN(COLLID('daily'))")
```

From the OS/390 operator console:

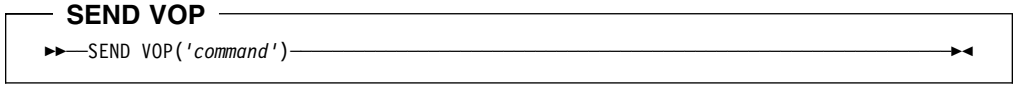
```
F taskname,SEND SRP('DROP ECAMDEVICE(220)')
```

SEND VOP

* ***Subcommand of SIBBATCH or SIBADMIN***
* ***System Operator command***

The SEND VOP subcommand sends a command request to the console operator interface (virtual operator panel) task.

SEND VOP



Values: *command* is any valid command (1 to 127 characters) enclosed in matching quotes (single or double).

Default value: None

Examples: From SIBBATCH or SIBADMIN:

```
SEND VOP('VARY DEVICE (UNIT (345) DISABLE)')
```

From the OS/390 operator console:

```
F taskname,SEND VOP('V DEV (UNIT (345) DIS)')
```

SET Subcommands

The SET subcommands control the setting or define the value of SVAA variables. The default values shown in the subcommand descriptions are the values that are in effect if you have not explicitly defined them with the SET subcommand.

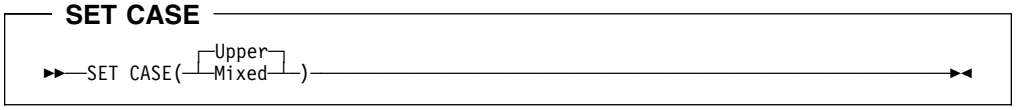
Parameter values set with the SET subcommands are not saved from one invocation of SIBADMIN or SIBBATCH to the next.

SET subcommands issued under SIBADMIN apply only to the current execution of SIBADMIN (and SIBBATCH does not have access to the variables set under SIBADMIN). Issue SET subcommands for SIBBATCH in a REXX EXEC, or issue them in either the SIBSTK00 parmlib member or in another parmlib member specified with the =PARMLIB parameter of the SIBBATCH command.

SET CASE

***** ***Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand specifies whether values of parameters that normally allow lowercase characters are to be translated automatically to uppercase.



Values: **Upper** or **Mixed**

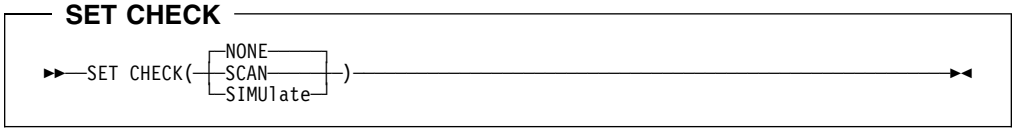
Default value: **U**

Example: **SET CASE(M)**

SET CHECK

***** ***Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand specifies whether SVAA subcommand syntax checking or simulation—or neither one—is to be performed.



Values: **NONE** turns off both subcommand syntax checking and simulation processing.

SCAN turns on subcommand syntax checking. This option enables you to validate syntax for an entire batch file before it is executed.

SIMU turns on simulation processing of the DDSR facility. After you issue **SET CHECK(SIMU)**, DDSR performs all space release functions except for actually releasing space; that is, it does no I/O to the subsystem. Turn simulation off with **SET CHECK(NONE)**.

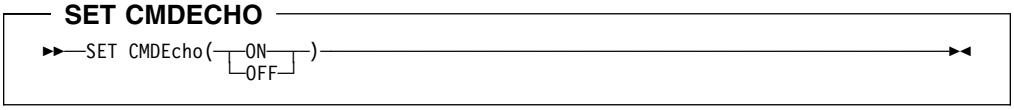
Default value: **NONE**

Example: **SET CHECK(SCAN)**

SET CMDECHO

* ***Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand specifies whether SVAA subcommands are to be displayed at the destination specified by **SET DEST OUTMSG** before execution.



Abbreviation: **SET CMDE**

Values: **ON** or **OFF**

Default value: **ON** for SIBBATCH and SIBSRP
OFF for SIBADMIN

Example: **SET CMDE(ON)**

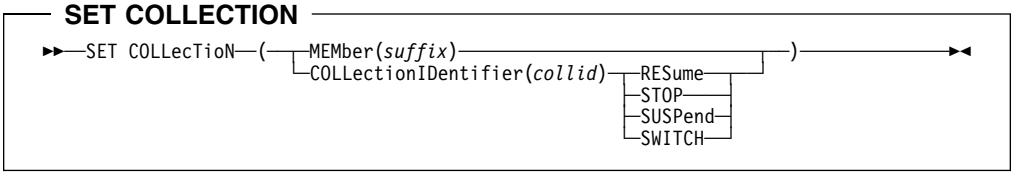
SET COLLECTION

* ***Subcommand of SIBADMIN, SIBBATCH , or SIBSRP***
* ***System Operator command***

Use this subcommand or command to modify collection-run parameters.

This subcommand enables you to:

- Resume data collection after a SET COLLTN SUSPEND subcommand. (RESUME parameter)
- Specify that the collection run identified in the COLLID parameter is to be stopped. Reporter stops the collection run and, if no other collection run is using the output file, closes the output file. (STOP parameter)
- Halt data collection temporarily without terminating the Data Collection Task; it eliminates the need to re-establish control settings. (SUSPEND parameter)
- Switch between the main and alternate logging files explicitly through this subcommand. (SWITCH parameter)



Parameters

COLLECTIONIDENTIFIER (collid)

Conditional. You must specify either COLLECTIONIDENTIFIER or MEMBER.

Identifies the collection run to be acted upon.

Abbreviation: **COLLID**

Values: *collid* is the identifier of the collection run (defined in the COLLECT ORDATA subcommand).

If the identifier contains lowercase alphabetic characters and SET

CASE(UPPER) is in effect, you must enclose the identifier in matching quotes.

Default value: N/A

MEMBER (suffix)

Conditional. You must specify either COLLECTIONIDENTIFIER or MEMBER.

This parameter specifies the suffix of the member (SIBSRPnn) in the PARMLIB that contains the new data-collection parameters. Reporter appends the suffix to the SIBSRP string to produce the name of the member to be accessed in the library specified by the STKPARMS DD statement. See "Starting the Data Collection Task" on page A-2 for details of the STKPARMS DD statement.

The net result of specifying this parameter is as if you had specified COLL ORD or COLL PT with the MODIFY(YES) parameter. The collection identifier in the COLL subcommand in the specified member must identify a currently running collection.

Abbreviation: `MEM`

Values: *suffix* specifies the two-character suffix of the STKSRP*nn* member.

Default value: N/A

RESUME

Conditional. If you specify COLLECTIONIDENTIFIER, you must follow it with the RESUME, STOP, SUSPEND, or SWITCH parameter.

Specifies that data collection is to be resumed. This parameter enables you to resume data collection after a SET COLLECTION SUSPEND subcommand.

Abbreviation: `RES`

Values: None (keyword parameter only)

STOP

Conditional. If you specify COLLECTIONIDENTIFIER, you must follow it with the RESUME, STOP, SUSPEND, or SWITCH parameter.

Specifies that data collection is to be stopped.

Abbreviation: None

Value: None (keyword parameter only)

SUSPEND

Conditional. If you specify COLLECTIONIDENTIFIER, you must follow it with the RESUME, STOP, SUSPEND, or SWITCH parameter.

Specifies that data collection is to be suspended. The output file is not closed.

Reporter checks user authority if file switching is not initiated at an operator console (see "Security Checking" on page 3-10 for details).

Abbreviation: `SUSP`

Values: None (keyword parameter only)

SWITCH

Conditional. If you specify COLLECTIONIDENTIFIER, you must follow it with the RESUME, STOP, SUSPEND, or SWITCH parameter.

Specifies that the current logging file (main or alternate) be switched to the other file. If there is a switch exit, control is given to the exit for further action. Options in the exit are to close the current logging file, switch to the other file, or, if there is no alternate file, start writing at the beginning of the current logging file. For more information, see “Handling Full Logging Files” on page 5-5.

File-switching subcommands must include the collection run identifier. Switching the logging file invokes the logging-file-full routine (see “Handling Full Logging Files” on page 5-5), and may result in the files being switched if you have specified an alternate file. If the data-collection run specified by *collid* is writing output to SMF, Reporter rejects this subcommand.

Abbreviation: None

Values: None (keyword parameter only)

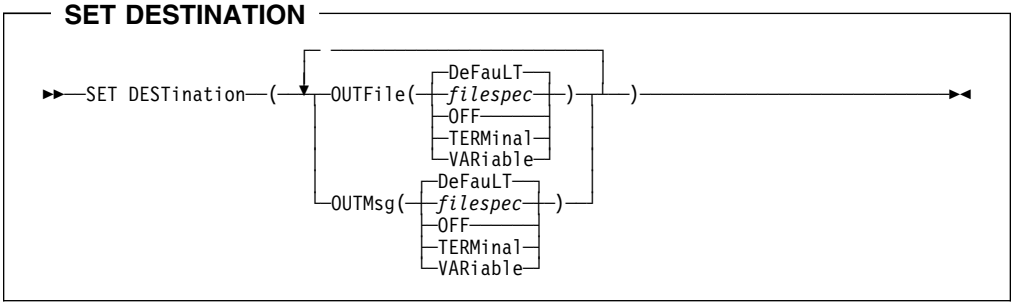
SET DESTINATION

* ***Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand enables you to redirect messages (via **OUTMsg**) and reports (via **OUTFile**) to alternate destinations. SET DESTINATION does not affect the destination of:

- SAS reports (Device Performance, Cache Effectiveness, and Space Utilization reports)
- Command output or messages generated by ISPF requests (you must use REXX variables)
- Messages normally sent to SIBLOG, such as Interval DDSR messages

Note: You can redirect the output of any individual SVAA subcommand, thus overriding (for *only* that subcommand) the destination you specify with SET DESTINATION. For more information, see “Redirecting Subcommand Output” on page 8-10.



Abbreviation: **SET DEST**

Values: **OUTF** redirects report output (for example, LISTCFG subcommand output) to the destination indicated by the subparameter value.

OUTM redirects a message to the destination indicated by the subparameter value.

The subparameter is one of the following:

filespec Direct output to the file *filespec* (a SAS/C-style file name).

filespec has a maximum of 48 characters.

You can use the style prefix (see "Using the Style Prefix" on page 8-8 for more information).

Note: In the TSO environment, follow standard data set naming conventions; that is, enclose a data set name in quotes if it is fully qualified.

DeFauLT In OS/390 batch, **OUTFile** defaults to the destination specified by the SYSPRINT DD name. **OUTMsg** defaults to the destination specified by the SYSTERM DD name.

In TSO, both destinations default to **TERMINAL**.

In ISPF, both destinations default to **VARIABLE**.

OFF Suppress output.

TERMinal Display output at the terminal.

VARiable Direct output to *exec* stem variables (not valid in SIBBATCH environment).

SVAA output for destination OUTMSG is written to stem variable SIBLMSG.*n*. This allows a REXX EXEC to issue an SVAA subcommand and analyze the returned results (messages and reports) in REXX variables.

Output for destination OUTFILE is written to the stem variable SIBRMSG.*n* (*n* is the value of an occurrence of SIBLMSG or SIBRMSG).

The variables with *n*=0 contain the number of variables generated. For example, if SIBRMSG.0=17, then variables SIBRMSG.1 through SIBRMSG.17 contain the data. The SIBRMSG*n* and SIBLMSG*n* variables are set to 0 before the execution of each SVAA subcommand.

Note: REXX must be available to use this value.

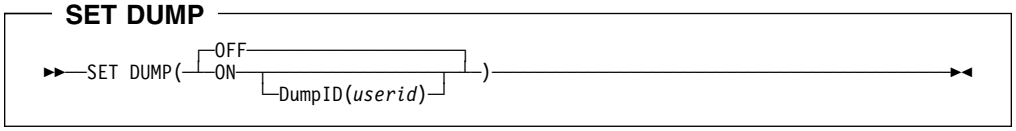
Default value: **DFLT**

Example: **SET DEST(OUTFILE(ddn:systst) OUTMSG(VAR))**

SET DUMP

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand specifies whether a dump is to be produced if SVAA terminates abnormally. Additionally, you have the option of specifying a TSO userid to be notified that the abend occurred.



Values: **ON** or **OFF** and optionally: **DID(***userid***)**

userid is 1 to 8 characters

Default value: **OFF**

userid defaults to the user executing the subcommand.

Example: **SET DUMP(ON DID(MAINT9))**

SET ECAMDEVICE

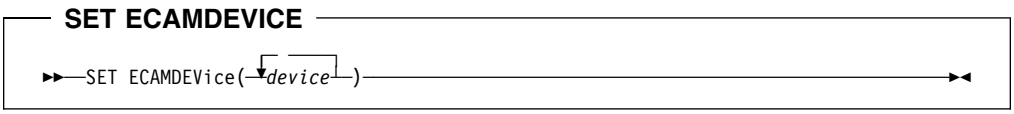
* ***Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand specifies devices (as defined to the host) to be used for ECAM communications. You can specify multiple (up to 100) ECAM devices.

You must include the SET ECAMDEVICE subcommand in the PROFSIBA REXX EXEC (if you are using it), and in any of the following parmlib members you are using: SIBSTK, SIBSFC, SIBSIM, SIBSRP, and SIBVOP.

Notes:

1. Do not specify a device that does not exist or has not been mapped.
2. If you issue this subcommand while the target device is offline, the device is considered inactive and does not appear in response to a QUERY ECAMDEVICE (ALL) subcommand.
3. Do not issue this subcommand against a PPRC primary or secondary volume; that is, do not try to define a PPRC volume as an ECAM device.
4. You cannot use this subcommand on PAV Alias or Power PPRC Bridge devices.



Abbreviation: **SET ECAMDEV**

Values: *device* is a host device number (1 to 4 hexadecimal digits). The device may be offline to OS/390.

You can specify both lists and ranges of devices.

Default value: None

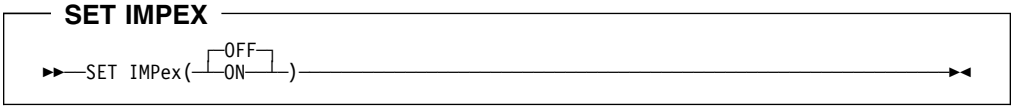
Example: **SET ECAMDEV(2aE 730 112D)**

SET IMPEX

* ***Subcommand of SIBADMIN or SIBSRP***

This subcommand specifies the setting for the implied execution of SVAA macros. If IMPEX is **ON**, an SVAA subcommand is treated as a potential SVAA macro. If the macro does not exist, normal SVAA subcommand processing is performed.

Note: If you specify **IMPEX(ON)**, there should not be any members in the CLIST concatenation with the same name as any SVAA subcommand or its abbreviation.



Abbreviation: **SET IMP**

Values: **ON** or **OFF**

Default value: **OFF**

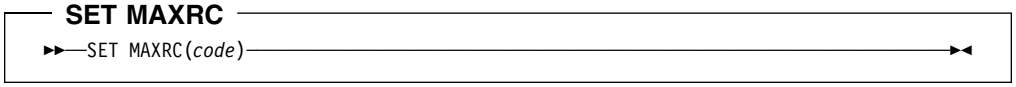
Example: **SET IMPEX(OFF)**

SET MAXRC

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

For SVAA initialization, this subcommand sets the maximum return code (severity) that will be accepted from the initialization routines and the SVAA profile macro and still allow initialization to continue.

SIBBATCH also uses MAXRC to specify the maximum return code value SVAA tolerates from a subcommand. Any return code that exceeds the MAXRC value causes SIBBATCH to terminate processing and ignore any remaining subcommands in the input file.



Values: *code* is an unsigned integer ≥ 0.

Default value: **8**

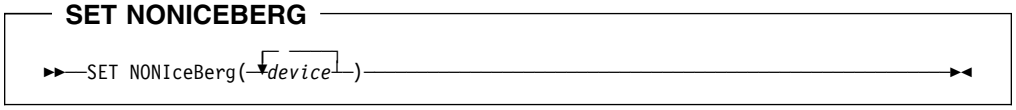
Example: **SET MAXRC(12)**

SET NONICEBERG

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand specifies devices (as defined to the host) over which non-SVA subsystem statistics are obtained.

Note: If you issue this subcommand while the target device is offline, the device is considered inactive and does not appear in response to a QUERY NONICEBERG (ALL) subcommand.



Abbreviation: **SET NONIB**

Values: *device* is a host device number (1 to 4 hexadecimal digits). The device may be offline to OS/390.

You can specify devices with both lists and ranges.

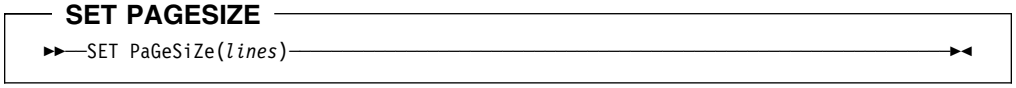
Default value: None

Example: **SET NONIB(333 2aa)**

SET PAGESIZE

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand specifies the number of lines per page for SVAA reports.



Abbreviation: **SET PGSZ**

Values: *lines* is a 2-digit number ranging from 40 to 99.

Default value: **60**

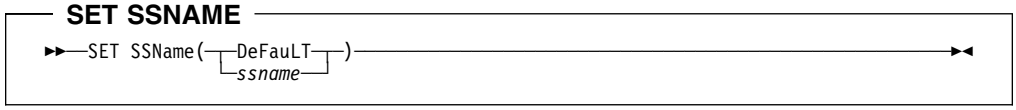
Example: **SET PGSZ(55)**

SET SSNAME

*** *Subcommand of SIBBATCH or SIBADMIN***

This subcommand identifies the OS/390 subsystem that is to perform SVAA services (that is, the subsystem to which subsequent SVAA commands will be directed). You would normally use this subcommand only to perform installation verification after installing a new release of SVAA.

SET SSNAME



Abbreviation: **SET SSN**

Values: *ssname* is the name of the OS/390 subsystem that is to perform SVAA services.

DEFAULT (or **DFLT**) sets *ssname* to the name of the first activated OS/390 subsystem running SVAA.

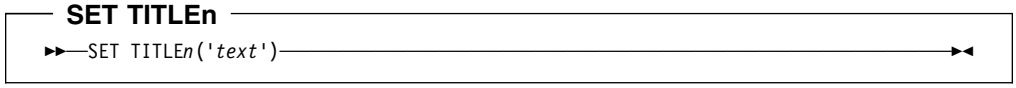
Example: **SET SSN(IBM3)**

SET TITLEn

*** *Subcommand of SIBBATCH, SIBADMIN, or SIBSRP***

This subcommand (**SET TITLE1**, **SET TITLE2**, or **SET TITLE3**) defines the text string (such as an organization name or address) that is to appear in one of the three title lines of SVAA reports. *n* is **1** (line 1), **2** (line 2), or **3** (line 3). SVAA converts all titles to uppercase.

SET TITLEn



Values: *text* is a text string (0 to 127 characters).

The text must be enclosed in parentheses and either single or double quotation marks. However, if the text string contains no blanks, the quotation marks are optional.

Default value: None

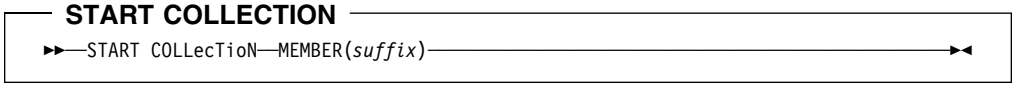
Example: **SET TITLE1('Line 1 of report title')**

START COLLECTION Command

* ***Subcommand of SIBADMIN, SIBBATCH, or SIBSRP***
* ***System Operator command***

The START COLLECTION command starts a collection run.

START COLLECTION



MEMBER (suffix)

Required

This parameter specifies the suffix of the member (SIBSRPnn) in the PARMLIB that starts data collection. Reporter appends the suffix to the SIBSRP string to produce the name of the member to be accessed in the library specified by the STKPARMS DD statement. The collection identifier specified in the STKPARMS member must not already have been established.

Abbreviation: **MEM**

Values: *suffix* is the 1- or 2-digit member suffix.

START Commands

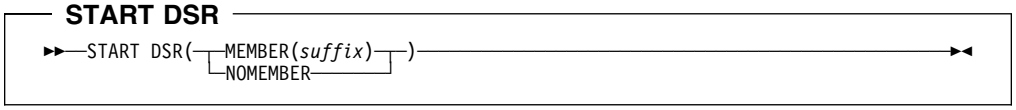
The START commands are issued only from an OS/390 system console. They enable you to start tasks in the SVAA address space.

START DSR

* ***System Operator command***

The START DSR command initializes the DDSR Main Task, which in turn initializes DDSR Interval tasks or the Dynamic DDSR intercept. If the DDSR Main Task is already running, use the START RELEASE subcommand to start DDSR Interval tasks.

START DSR



Note: The following parameter description applies to all four START commands.

MEMBER (suffix) | NOMEMBER

Optional

This parameter specifies the suffix of the parmlib member to be used when initializing the specified task—or that the task is to be initialized with no parmlib member.

Abbreviation: **MEM** or **NOMEM**

Values: *suffix* consists of 1 to 2 alphanumeric characters in the set: A to Z, 0 to 9, $, #, @

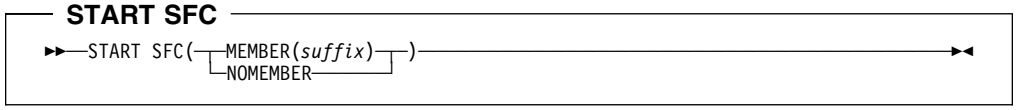
Default value: If you specify neither **MEM** nor **NOMEM**, the parmlib member suffix defaults to **00**.

START SFC

*** *System Operator command***

The START SFC command initializes the Server Framework Component (SFC) task, which in turn initializes the SVAA Server address space.

START SFC



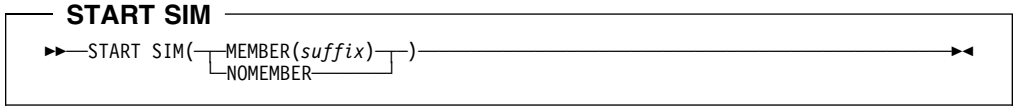
See "START DSR" on page 8-75 for a description of the MEMBER and NOMEMBER parameters.

START SIM

*** *System Operator command***

The START SIM command initializes the Service Information Message (SIM) task.

START SIM



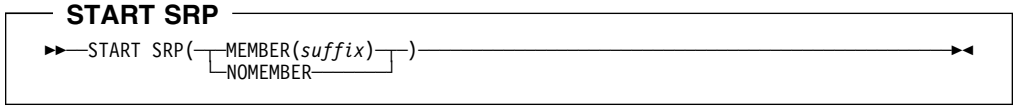
See "START DSR" on page 8-75 for a description of the MEMBER and NOMEMBER parameters.

START SRP

*** *System Operator command***

The START SRP command initializes the Reporter Data Collection task.

START SRP



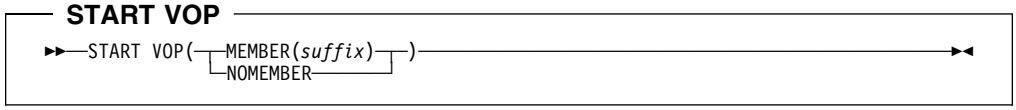
See "START DSR" on page 8-75 for a description of the MEMBER and NOMEMBER parameters.

START VOP

*** *System Operator command***

The START VOP command initializes the console operator interface (virtual operator panel) task.

START VOP



See "START DSR" on page 8-75 for a description of the MEMBER and NOMEMBER parameters.

STOP Commands

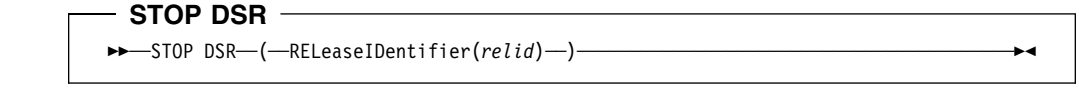
The STOP commands are issued only from an OS/390 system console. They enable you to terminate tasks that are running in the SVAA address space.

STOP DSR

*** *System Operator command***

The STOP DSR command stops all DDSR Interval tasks and the Dynamic DDSR intercept, and then terminates the DDSR Main Task.

STOP DSR



Values: *relid* is the release task identifier for this DDSR task. You can use widcard characters but not ranges or lists.

Note: If **SET CASE(UPPER)** is in effect, and you use lowercase characters in the release task identifier and don't want them converted to uppercase, you must enclose the identifier in matching quotes.

Default value: None

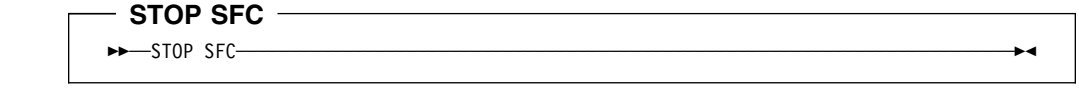
Example: **RELID('task011')**

STOP SFC

*** *System Operator command***

The STOP SFC command stops the Server Framework Component (SFC) task.

STOP SFC

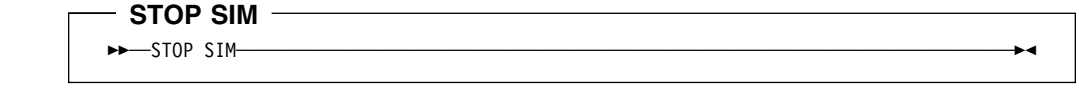


STOP SIM

*** *System Operator command***

The STOP SIM command stops the Service Information Message (SIM) task.

STOP SIM

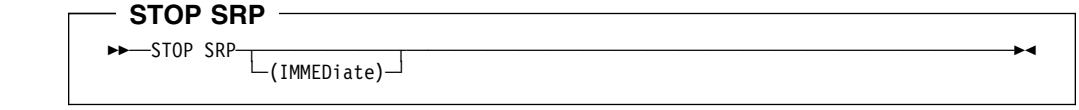


STOP SRP

*** *System Operator command***

The STOP SRP command stops the Reporter Data Collection task. Reporter collects a final set of statistics for each outstanding collection run, unless you specify the IMMED parameter.

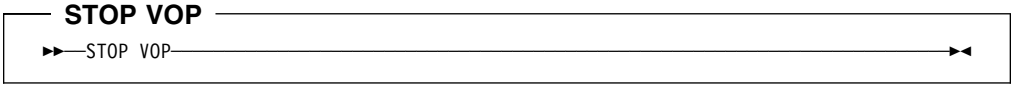
STOP SRP



STOP VOP

* ***System Operator command***

The STOP VOP command stops the console operator interface (virtual operator panel) task.



SUMMARIZE HISTORYDATA Subcommand

*** *Subcommand of SIBBATCH or SIBADMIN***

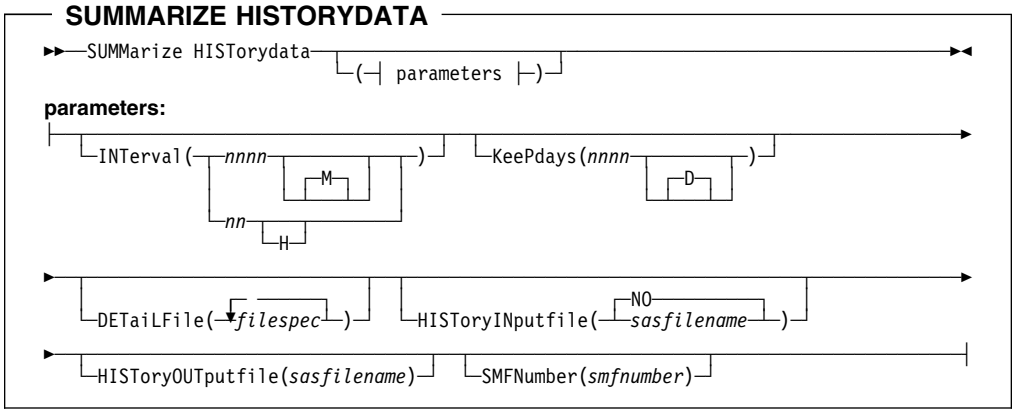
This subcommand summarizes detail PT data in the Reporter history files. The abbreviation is SUMM HIST.

Data accumulates in four files collectively called the history file, for input to the performance trend analysis reports. The history file contains summarized data for the number of days specified by the KP parameter.

For history file summarization, you can specify:

- The periods of time covered by each report statistic (INTERVAL parameter).
- The number of day's collections to keep in the history file after summarization (KEEPDAYS parameter).
- The names of the logging files to be summarized (DETAILFILE parameter).
- The names of any previously unsummarized files to be summarized (HISTORYINPUTFILE parameter).
- The name of the SAS data set to be used for the summarized output (HISTORYOUTPUTFILE parameter)
- The SMF record type for your collection records (SMFNUMBER parameter).

Note: Before using the SUMM HIST subcommand, you must allocate a data set with a DD of SASPGM.



Parameters

All the parameters are optional. If you do not specify any parameters, Reporter uses the default value of all parameters.

See page 8-82 for examples of coding the parameters.

DETAILFILE (filespec)

Optional

Specifies the names of any number of unsummarized logging files to be used as input to the summarization processes.

Abbreviation: `DETLF`

Value: *filespec* specifies the detail input files.

If you use a style prefix, it must be DDN (see "Using the Style Prefix" on page 8-8 for more information).

Default value: **DDN:DETAILIN**, which must identify a DD statement in the SAS step of the Summarize History Data JCL.

HISTORYINPUTFILE (sasfilename | NO)

Optional

Specifies the name of the input (previously summarized) history file.

Abbreviation: **HISTIN**

Values: *sasfilename* specifies the input history file. It cannot exceed 7 characters.

The *sasfilename* must be different from that of the HISTOUT (output history) file.

NO specifies that there is no input history file (that is, this is the first summarization run).

Default value: **STKPERFI.HISTORY**

Notes:

1. If this is the first history data summarization run, specify **NO** to create a new history data summary file.
2. Unless you specify **NO**, this parameter must specify a SAS data set. See "Specifying SAS Data Sets" on page 8-9 for more information.
3. The recommended procedure is for the SAS filename part of both the HISTIN and HISTOUT sasfilenames to be the same, and for the DDNAME part of the sasfilenames to refer to separate DD statements that refer to the same generation data group. For example:

   ```
   HISTIN(STKPERFI.HISTORY)
   HISTOUT(STKPERFO.HISTORY)
   ```

   For a JCL example, see "Summarizing the History File" on page A-3.
4. If you are appending to an existing history file, a return code of 4 is normal.

HISTORYOUTPUTFILE (sasfilename)

Optional

Specifies the name of the output history file, which must be a SAS data set. See "Specifying SAS Data Sets" on page 8-9 for rules for specifying SAS data sets.

Abbreviation: **HISTOUT**

Value: *sasfilename* specifies the output history file. It cannot exceed 7 characters.

The *sasfilename* must be different from that of the HISTIN (input history) file.

Default value: **STKPERFO.HISTORY**

Note: The recommended procedure is for the SAS filename part of both the HISTIN and HISTOUT sasfilenames to be the same, and for the DDNAME part of the

sasfilenames to refer to separate DD statements that refer to the same generation data group. For example:

```
HISTIN(STKPERFI.HISTORY)
HISTOUT(STKPERFO.HISTORY)
```

For a JCL example, see “Summarizing the History File” on page A-3.

INTERVAL (nnnnM | nnH)

Optional

Specifies the data-summarization interval, which:

- Must be a multiple of the original data-summarization interval
- Must be a factor of either 1440 or 24.

If the history data-summary file already exists, this parameter must be a multiple of the originally specified summarization interval.

Abbreviation: **INT**

Values: *nnnn***M** specifies the interval as a number of minutes, 1 to 1440, that must be exactly divisible into 1440. Leading zeroes are allowed. You cannot specify 0.

*nn***H** specifies the interval as a number of hours, 1 to 24, that must be exactly divisible into 24. A leading zero is allowed. You cannot specify 0.

Default value: 120 minutes.

KEEPDAYS (nnnnD)

Optional

Specifies the number of days' collections to keep in the history file after summarization.

Abbreviation: **KP**

Values: *nnnnD* specifies the number of days' collections to keep, where *nnnn* is a positive number. Leading zeros are allowed. You cannot specify **0**.

Default value: If you omit this parameter, 395 days (approximately 13 months) of data is kept.

SMFNUMBER (smfnum)

Conditional

This parameter, which is usually not needed, specifies the SMF record type assigned by your system programmer for your data collection records in SMF.

The SMF record type is normally specified to Reporter using the SUBPARM parameter in the SMFPRMxx member of SYS1.PARMLIB. This parameter is usually optional, but must be specified if the input file contains SMF records and if one or more of the following is true:

- The SMF data was created on another host with a different SVAA SMF number (as when you are producing the report on a VM system but the data was collected on an OS/390 system).
- The OS/390 host on which the data is being processed does not contain an entry for SVAA in the SIBPRMxx member.

- The OS/390 host on which the data is being processed does not have an initialized and active SVAA subsystem.

Abbreviation: **SMFN**

Values: *smfnum* specifies SMF record type as a whole number ranging from 128 to 255 inclusive.

Default value: None.

Examples of SUMM HIST

Example 1.

```
SUMM HIST
```

This example requests a history data summary with all defaults. Data is to be summarized at two hour intervals. The detail file is pointed to by the DETAILIN DD statement in the JCL. The input history file is in the SAS library pointed to by the STKPERFI DD statement in the JCL. The input SAS file name is HISTORY. The new summarized history file output is in the SAS library pointed to by the STKPERFO DD statement in the JCL. The summarized SAS file name is HISTORY.

Example 2.

```
SUMM HIST-
    (INT(240M)-
    KP(210D)-
    DETLF(DDN:RPTDETL)-
    HISTIN(NO)-
    HISTOUT(STKPERF.NEWHIST) )
```

Data is to be summarized at four-hour intervals. The history file is to contain data for the last 210 days (seven months). The detail file has DD name RPTDETL. The history output file is being created the first time; therefore, there is no history input file. The new summarized history file is in the SAS library pointed to by the STKPERF DD statement in the JCL. The SAS file name is NEWHIST.

Chapter 9. Understanding the SVAA Reports

Chapter Summary

This chapter describes the layout of the default SAS reports of the performance and space utilization of your subsystems. The performance report examples are from the interval Device Performance and Cache Effectiveness reports; the time-ordered and summary reports have the same format as the interval reports.

"Definitions of Field Names" on page 9-15 defines the field names and column headings used in the reports.

Each printed page of any report has a common header composed of up to three title lines, a date-and-time stamp, and a page number:

```
                                Title1                          12:58 Wednesday, November 18, 1998  1
                                Title2
                                Title3
```

Title1, Title2, and **Title3** are the title lines you specify for the report in one of these ways:

- Using SET TITLE subcommands in the SIBSTK00 member of SYS1.PARMLIB. SVAA uses this member during SVAA initialization.
- Using SET TITLE subcommands in a SIBADMIN session.
- Using the **Session Administration** function of the SVAA panels.

You can specify up to three title lines in this manner; none are required. The lines you specify (up to 127 characters in each) appear centered at the top of each page of your reports:

SAS inserts the mixed-case date-and-time stamp, followed by the page number, at the right of the first title where there is room. The time is the time that SAS was invoked to produce the report. If there is no room for the stamp on the title lines, it appears on the next available line. SAS inserts a blank line after the date-and-time stamp line.

Note: In most of the examples in this chapter, the title lines and data-and-time stamp are omitted.

Reading the Selected Input Record Summary Report

This report appears automatically whenever you request a Device Performance report and whenever you issue a SUMM HIST subcommand. The report can indicate that there might be duplicate data due to collection being made from more than one host. It is important that collection be made from one host by specifying different collection identifiers for data being collected on different hosts. You should decide which host to collect data on.

The report has the following format:

```
                                                REPORT TITLE                        13:29 Tuesday, July 27, 1999
                                               REPORT SUB-TITLE
REPORTER                                SELECTED INPUT RECORDS SUMMARY REPORT        27JUL1999    13:29:50
                                                                                    SIBIRSR  V1 R0 L1
  SUBSYSTEM  COLLECTION ID     SMF ID  CPU ID    EARLIEST RECORD   LATEST RECORD
                                                 TIME STAMP        TIME STAMP
  ---------  ----------------  ------- --------  ----------------  ----------------
  ISPROD10   PERFORMANCETRACK  MVSB    MVSB      05JUL99:23:00:00  07JUL99:00:00:00
* ISPROD10   PERFORMANCETRACK  MVSA    MVSA      06JUL99:04:44:00  06JUL99:21:14:00
  ISPROD40   PERFORMANCETRACK  MVSB    MVSB      05JUL99:23:00:00  07JUL99:00:00:02
* ISPROD40   PERFORMANCETRACK  MVSA    MVSA      06JUL99:04:44:00  06JUL99:21:14:00
  ISPROD50   PERFORMANCETRACK  MVSA    MVSA      06JUL99:00:00:01  07JUL99:00:00:02
*** WARNING! POTENTIAL OVERLAPPING OF DATA COLLECTION INTERVALS DETECTED.
    DUPLICATE DATA MAY BE IN THE INPUT FILE. PLEASE VERIFY THAT MULTIPLE
    REPORTER DATA COLLECTION TASKS, E.G. RUNNING ON DIFFERENT HOSTS,
    ARE NOT COLLECTING DATA FOR THE SAME SUBSYSTEM(S).
```

In this report, the asterisk at the left indicates a subsystem and collection identifier for which there may be duplicate data.

Reading Device Performance Reports

This section describes the layout of the default Device Performance reports.

Report-Specific Header Lines

In addition to the common header described above, each page of a Device Performance Interval Report has the following heading:

```
REPORTER                              DEVICE PERFORMANCE INTERVAL REPORT                  25OCT1999    15:20:38
                                                                                         SIBDP10  V1 R0 L1
SUBSYSTEM STK          INTERVAL ENDING: 10OCT1999   8:29 INTERVAL DURATION:  0:15

FDID   DEV   VOLSER   T/P  DEVICE   % DEV    I/O    KBYTES   ACCESS    -I/O SERVICE TIME (MS)- % DEV  % DEV  % DEV
      ADDR                TYPE     AVAIL  PER SEC  PER SEC  DENSITY   TOTAL   DISC   CONNECT   UTIL   DISC   CONN
----  ----  --------  ---  ------   -----  -------  -------  -------   -----  ------  -------  -----  -----  -----
```

These lines show:

1. The type of report and the date and time that SAS created the report.
2. The version, release, and level of SVAA.
3. The SAS macro that produced the report.
4. The name of the subsystem for which the report was run, the date and time of the end of the interval being reported on, and the duration of each interval.
5. The field titles or names of the columns for the report. See “Definitions of Field Names” on page 9-15 for an explanation of the field names.

Sample Device Performance Report

The listing below shows a sample Device Performance Interval Report. The following subcommand produced the report.

```
REPORT PERF-
    (DEVPERF(YES)-
    INTRPT(YES)-
    DEVDET(YES)-
    OVERALL(YES)-
    DATES(10OCT1999)-
    TIMES(0800:1200)-
    RPTINT(15M) )
```

The largest section of a Device Performance Interval Report is composed of sections like the following example, one for each of the intervals in the specified time range. There is one line in this section for each functional device in the collected data, because neither the **UNIT** nor **VOLUME** parameter was specified.

If you specify **DEVDET(NO)**, the detailed device lines do not appear in the report.

```
REPORTER                              DEVICE PERFORMANCE INTERVAL REPORT                  25OCT1999    15:20:38
                                                                                         SIBDPIO  V1 R0 L1
SUBSYSTEM STK          INTERVAL ENDING: 10OCT1999   8:29 INTERVAL DURATION:  0:15

FDID   DEV   VOLSER   T/P  DEVICE   % DEV    I/O    KBYTES   ACCESS    -I/O SERVICE TIME (MS)- % DEV  % DEV  % DEV
       ADDR                TYPE     AVAIL  PER SEC  PER SEC  DENSITY   TOTAL   DISC   CONNECT   UTIL   DISC   CONN
----   ----  --------  ---  ------   -----  -------  -------  -------   -----  ------  -------  -----  -----  -----
0000   0400  ICE400     P   33901    100.0      0.0      0.0      0.0    22.2     0.1     22.1    0.0    0.0    0.0
0001   0401  PRDV02     P   33901    100.0      0.7      2.7      0.4    10.2     8.0      2.3    0.7    0.5    0.2
0002   0402  PRDV03     P   33901    100.0      0.1      0.2      0.0     8.1     6.2      1.9    0.1    0.1    0.0
0003   0403  PRDV04     P   33901    100.0      0.8      5.0      0.4    10.5     7.0      3.5    0.8    0.5    0.3
0004   0404  PRDV05     P   33901    100.0      0.3      0.8      0.2     9.7     7.8      2.0    0.3    0.3    0.1
0005   0405  PRDV06     P   33901    100.0      0.0      0.6      0.0    26.9    20.4      6.5    0.1    0.1    0.0
0006   0406  PRDV07     P   33901    100.0      0.1      1.0      0.1    13.8    10.1      3.8    0.2    0.1    0.0
0007   0407  PRDV08     P   33901    100.0      0.1      0.1      0.0    11.7     9.4      2.3    0.1    0.1    0.0
0008   0408  PRDV09     P   33903    100.0      0.1      0.1      0.0     8.7     6.4      2.3    0.1    0.0    0.0
0009   0409  PRDV10     P   33903    100.0      6.1    131.9      3.2    12.4     4.2      8.2    7.5    2.6    5.0
000A   040A  PRDV11     P   33903    100.0      1.1      4.8      0.6     5.6     2.8      2.8    0.6    0.3    0.3
000B   040B  PRDV12     P   33903    100.0      0.1      0.2      0.1     6.6     4.6      2.0    0.1    0.1    0.0
000C   040C  PRDV13     P   33903    100.0      2.6      9.8      1.4     4.7     2.7      2.0    1.2    0.7    0.5
000D   040D  PRDV14     P   33903    100.0      1.4      4.9      0.7     6.2     4.1      2.0    0.9    0.6    0.3
000E   040E  PRDV15     P   33903    100.0      0.0      0.1      0.0    11.7     9.9      1.8    0.0    0.0    0.0
  ⋮
00FD   04FD  ORVM14     P   33909    100.0      2.0     11.5      0.7     7.2     4.6      2.7    1.5    0.9    0.5
00FE   04FE  ORVM15     P   33909    100.0      0.2      1.7      0.1    15.8    12.2      3.5    0.3    0.3    0.1
00FF   04FF  ORVM16     P   33909    100.0      1.4      6.5      0.5    13.7    10.9      2.8    1.9    1.5    0.4
================================================================================================================================

SELECTED DEVICES SUMMARY

      SUBSYSTEM            % DEV    I/O    KBYTES   ACCESS    -I/O SERVICE TIME (MS)- % DEV  % DEV  % DEV
       SUMMARY             AVAIL  PER SEC  PER SEC  DENSITY   TOTAL   DISC   CONNECT   UTIL   DISC   CONN
                           -----  -------  -------  -------   -----  ------  -------  -----  -----  -----
      PROD PARTITION       100.0    352.3   2917.4      0.7     7.9     3.5      4.4    1.1    0.5    0.6
      OVERALL TOTALS       100.0    352.3   2917.4      0.7     7.9     3.5      4.4    1.1    0.5    0.6

SELECTED SCSI DEVICES SUMMARY

      SUBSYSTEM            % DEV    I/O    KBYTES   ACCESS    -I/O SERVICE TIME (MS)- % DEV  % DEV  % DEV
       SUMMARY             AVAIL  PER SEC  PER SEC  DENSITY   TOTAL   DISC   CONNECT   UTIL   DISC   CONN
                           -----  -------  -------  -------   -----  ------  -------  -----  -----  -----
      PROD PARTITION       100.0      0.0      0.0      0.0     0.0     0.0      0.0    0.0    0.0    0.0
      OVERALL TOTALS       100.0      0.0      0.0      0.0     0.0     0.0      0.0    0.0    0.0    0.0
```

Note: The last three columns of the Device Performance Interval Report may show apparently anomalous data for devices that are designated for ECAM I/O (that is, for communication between Reporter and the SVA subsystem). Reporter issues a single, very long channel program to each subsystem, once each collection interval. Hence, there may be less than 0.0 I/Os per second over a 15-minute collection interval and at the same time there is a high connect time per I/O. Low values for % DEV UTIL and % DEV CONN may also be expected for ECAM devices, because this data is reported per device, not per I/O. For FICON channels, the value for % DEV CONN may exceed 100%.

Following the above report for each of the devices, the report for each interval contains three additional small sections:

- Disk array summary
- Channel interface performance
- Distribution of drive module utilization

Disk Array Summary

The **DISK ARRAY SUMMARY** section summarizes back-end space utilization and some features of back-end activity.

```
SUBSYSTEM STK                DISK ARRAY SUMMARY

          AVG % DRIVE  COEFF OF     NET CAPACITY LOAD %   FREE SPACE COLLECTION LOAD    COLL FREE SPC (%)     UNCOLL FREE SPC (%)
          MODULE UTIL  VARIATION    TEST   PROD  OVERALL     TEST   PROD  OVERALL      TEST   PROD  OVERALL   TEST   PROD  OVERALL
          -----------------------   -----  -----  -------    -----  -----  -------     -----  -----  -------  -----  -----  -------
               0.0         0          0.0   89.5    89.5       0.0    0.0     0.0        0.0   10.3    10.3     0.0    0.2     0.2
```

Channel Interface Performance Information

The **CHANNEL INTERFACE PERFORMANCE** section shows how the channel interface is performing.

```
CHANNEL INTERFACE     CLUSTER     INTERFACE    INTERFACE  CHANNEL    CHANNEL      I/O    % ACTIVE
   PERFORMANCE                       ID           NAME     SPEED      TYPE      PER SEC  ON CHNL
                      -------     ---------    ---------  -------  -----------  -------  --------
                         0           A                     20.0  ESCON            95.2      37.5
                         0           B                     20.0  ESCON             0.0       0.0
                         0           C                     20.0  ESCON             0.0       0.0
                         0           D                     20.0  ESCON             0.0       0.0
                         0           I                     40.0  SCSI              0.0       0.0
                         0           K                     40.0  SCSI              0.0       0.0
```

Distribution of Drive Module Utilization

The **DISTRIBUTION OF DRIVE MODULE UTILIZATION** section shows how evenly distributed is the back-end device utilization (the numbers should be reasonably even across the distribution).

```
=============================================================================================================================
                      |---------------------------|------|------|------|------|------|------|------|------|------|------|
                      | FREQUENCY (PERCENTILE)    |  10  |  20  |  30  |  40  |  50  |  60  |  70  |  80  |  90  | 100  |
DISTRIBUTION OF DRIVE |---------------------------|------|------|------|------|------|------|------|------|------|------|
MODULE UTILIZATION    | % DRIVE MODULE UTILIZATION|  5.3 |  5.4 |  5.7 |  5.8 |  6.0 |  6.1 |  6.2 |  6.3 |  6.6 |  6.7 |
                      |---------------------------|------|------|------|------|------|------|------|------|------|------|
=============================================================================================================================

 REPORT INCLUSION DAYS: ALL
 REPORT EXCLUSION DAYS: N/A
 REPORT EXCLUSION DATES: N/A
```

Overall Summary

After the sections for each of the intervals and because **OVERALL(YES)** was specified, the Device Performance Overall Summary appears. This summary is a weighted average of the statistics for the whole time range covered by the interval report.

The following example is a subsystem summary for the Production partition and for subsystem totals. The first portion of the Overall Summary report contains the same information as the device interval report. The summary portion contains the totals and averages for all the devices included in the device detail reports.

```
REPORTER                                DEVICE PERFORMANCE OVERALL SUMMARY                  25OCT1999    13:06:04
                                                                                           SIBDPIO  V1 R0 L1
SUBSYSTEM STK          REPORT START DATE: 10OCT1999      REPORT END DATE: 10OCT1999
                       REPORT START TIME:  8:14          REPORT END TIME: 11:59

FDID   DEV   VOLSER   T/P  DEVICE   % DEV    I/O    KBYTES   ACCESS    -I/O SERVICE TIME (MS)- % DEV  % DEV  % DEV
       ADDR                TYPE     AVAIL  PER SEC  PER SEC  DENSITY   TOTAL   DISC   CONNECT   UTIL   DISC   CONN
----  ----  --------  ---  ------   -----  -------  -------  -------   -----  ------  -------  -----  -----  -----
0000  0400  PRDV01     P   33901    100.0      0.0      0.0      0.0    27.4     4.6     22.7    0.0    0.0    0.0
0001  0401  PRDV02     P   33901    100.0      1.4      8.2      0.7     7.2     3.9      3.3    1.0    0.5    0.5
0002  0402  PRDV03     P   33901    100.0      0.6      1.1      0.3     4.2     2.0      2.3    0.3    0.1    0.1
0003  0403  PRDV04     P   33901    100.0      3.0     59.2      1.6    12.2     6.4      5.9    3.7    1.9    1.8
0004  0404  PRDV05     P   33901    100.0      0.1      0.3      0.1    10.5     7.9      2.6    0.1    0.1    0.0
0005  0405  PRDV06     P   33901    100.0      0.1      2.1      0.1    18.0     9.5      8.5    0.2    0.1    0.1
0006  0406  PRDV07     P   33901    100.0      0.2      1.0      0.1     9.9     6.6      3.3    0.2    0.1    0.1
0007  0407  PRDV08     P   33901    100.0      0.8     16.3      0.4    17.5    11.5      6.0    1.4    0.9    0.5
0008  0408  PRDV09     P   33903    100.0      0.1      1.0      0.1    11.5     6.1      5.4    0.1    0.1    0.1
0009  0409  PRDV10     P   33903    100.0      1.1     23.2      0.6    17.2    10.0      7.2    1.9    1.1    0.8
000A  040A  PRDV11     P   33903    100.0      0.5      1.8      0.2     6.2     3.4      2.8    0.3    0.2    0.1
000B  040B  PRDV12     P   33903    100.0      0.8     16.9      0.4    23.5    16.3      7.2    1.8    1.2    0.5
000C  040C  PRDV13     P   33903    100.0      1.6     12.0      0.8     8.2     5.1      3.1    1.3    0.8    0.5
000D  040D  PRDV14     P   33903    100.0      1.7     12.1      0.9     8.0     3.9      4.1    1.4    0.7    0.7
  ⋮
00FD  04FD  ORVM14     P   33909    100.0      0.9      9.1      0.3     7.3     3.2      4.0    0.7    0.3    0.4
00FE  04FE  ORVM15     P   33909    100.0      0.6      5.2      0.2     8.2     4.4      3.8    0.5    0.2    0.2
00FF  04FF  ORVM16     P   33909    100.0      2.1     13.7      0.7     7.9     4.4      3.5    1.6    0.9    0.7
```

In the summary section that follows, the only difference from the device-detail section of the report is the addition of the **SUBSYSTEM SUMMARY**, which shows, under the same field names, the totals and averages for all the devices included in the device-details report.

```
SUBSYSTEM STK          REPORT START DATE: 10MAY1999      REPORT END DATE: 10MAY1999
                       REPORT START TIME:  8:14          REPORT END TIME: 11:59
=============================================================================================================

     SUBSYSTEM           % DEV    I/O    KBYTES   ACCESS    -I/O SERVICE TIME (MS)- % DEV  % DEV  % DEV
      SUMMARY            AVAIL  PER SEC  PER SEC  DENSITY   TOTAL   DISC   CONNECT   UTIL   DISC   CONN
                         -----  -------  -------  -------   -----  ------  -------  -----  -----  -----
     PROD PARTITION      100.0    353.6   2562.4      0.7     7.5     4.0      3.6    1.0    0.5    0.5
     OVERALL TOTALS      100.0    353.6   2562.4      0.7     7.5     4.0      3.6    1.0    0.5    0.5

=============================================================================================================
SUBSYSTEM STK              DISK ARRAY SUMMARY

         AVG % DRIVE  COEFF OF     NET CAPACITY LOAD %   FREE SPACE COLLECTION LOAD    COLL FREE SPC (%)     UNCOLL FREE SPC (%)
         MODULE UTIL  VARIATION    TEST   PROD  OVERALL     TEST   PROD  OVERALL      TEST   PROD  OVERALL   TEST   PROD  OVERALL
         -----------------------   -----  -----  -------    -----  -----  -------     -----  -----  -------  -----  -----  -------
              0.0         0         0.0   89.5    89.5       0.0    0.0     0.0        0.0   10.3    10.3     0.0    0.2     0.2

=============================================================================================================
CHANNEL INTERFACE     CLUSTER     INTERFACE     INTERFACE  CHANNEL    CHANNEL      I/O    % ACTIVE
   PERFORMANCE                       ID           NAME     SPEED      TYPE      PER SEC  ON CHNL
                      -------     ---------    ---------  -------  -----------  -------  --------
                         0           A                      20.0  ESCON            94.1      30.3
                         0           B                      20.0  ESCON             0.0       0.0
                         0           C                      20.0  ESCON             0.0       0.0
                         0           D                      20.0  ESCON             0.0       0.0
                         0           M                      20.0  ESCON            94.2      30.3
                         0           N                      20.0  ESCON             0.0       0.0
                         0           O                      20.0  ESCON             0.0       0.0
                         0           P                      20.0  ESCON             0.0       0.0
                         1           A                      20.0  ESCON            99.1      30.5
                         1           B                      20.0  ESCON             0.8       0.0
                         1           C                      20.0  ESCON             0.0       0.0
                         1           D                      20.0  ESCON             0.0       0.0
                         1           M                      20.0  ESCON            99.7      30.0
                         1           N                      20.0  ESCON            42.5      19.8
                         1           O                      20.0  ESCON             0.0       0.0
                         1           P                      20.0  ESCON             0.0       0.0
=============================================================================================================
                     |-----------------------------|------|------|------|------|------|------|------|------|------|------|
                     | FREQUENCY (PERCENTILE)      |  10  |  20  |  30  |  40  |  50  |  60  |  70  |  80  |  90  | 100  |
DISTRIBUTION OF DRIVE|-----------------------------|------|------|------|------|------|------|------|------|------|------|
MODULE UTILIZATION    | % DRIVE MODULE UTILIZATION  | 5.4  | 5.5  | 5.5  | 5.6  | 5.6  | 5.7  | 5.7  | 5.8  | 5.8  | 6.0  |
                     |-----------------------------|------|------|------|------|------|------|------|------|------|------|
=============================================================================================================

 REPORT INCLUSION DAYS: ALL
 REPORT EXCLUSION DAYS: N/A
 REPORT EXCLUSION DATES: N/A
```

Reading Cache Effectiveness Reports

This section describes the layout of the default Cache Effectiveness reports. Each page has a common report header with title lines, date-and-time stamp and page number, as described at the beginning of this chapter.

Report-Specific Header Lines

In addition to the common header described on page 9-2, a Cache Effectiveness report has the following heading:

```
REPORTER                              CACHE EFFECTIVENESS INTERVAL REPORT                 16NOV1999    11:25:36
                                                                                         SIBCEIO  V1 R0 L1
INTERVAL ENDING: 10OCT1999   8:29     INTERVAL DURATION:   0:15

SUBSYSTEM NAME: STK         CACHE SIZE:  512 MB    NVS SIZE:  8 MB
                            OFFLINE CACHE:          0 BYTES     PINNED CACHE:          0 BYTES

FDID  DEV   VOLSER    T/P  DEVICE     READ    WRITE     I/O    READ    READ   WRITE   I/O    DFW    STAGE    HITS/    LOW
      ADDR                 TYPE     PER SEC  PER SEC  PER SEC  RATIO   HIT %  HIT %  HIT %  CONSTR  PER SEC   STGE  REF CT
----  ----  --------  ---  ------   -------  -------  -------  -----   -----  -----  -----  ------  -------  -----  ------
```

These lines show:

1. The type of report being printed and the time that SAS created the report.
2. The date and time the report interval ended and the duration of the interval.
3. The subsystem name, and its cache and nonvolatile storage sizes.
4. The field names and the headings of the columns in the report. See “Definitions of Field Names” on page 9-15 for an explanation of the field names.

Sample Cache Effectiveness Report

The report below is a sample interval Cache Effectiveness report for each of the devices. The following subcommand produced the report:

```
REPORT PERF-
     (CAEFF(YES)-
     INTRPT(YES)-
     DEVDET(YES)-
     OVERALL(YES)-
     DATES(10OCT1999)-
     TIMES(0800:1200)-
     RPTINT(15M) )
```

```
REPORTER                              CACHE EFFECTIVENESS INTERVAL REPORT                 16NOV1999    15:22:21
                                                                                         SIBCEIO  V1 R0 L1
INTERVAL ENDING: 10OCT1999   8:29     INTERVAL DURATION:   0:15

SUBSYSTEM NAME: STK         CACHE SIZE:  512 MB    NVS SIZE:  8 MB
                            OFFLINE CACHE:          0 BYTES     PINNED CACHE:          0 BYTES

FDID  DEV   VOLSER    T/P  DEVICE     READ    WRITE     I/O    READ    READ   WRITE   I/O    DFW    STAGE    HITS/    LOW
      ADDR                 TYPE     PER SEC  PER SEC  PER SEC  RATIO   HIT %  HIT %  HIT %  CONSTR  PER SEC   STGE  REF CT
----  ----  --------  ---  ------   -------  -------  -------  -----   -----  -----  -----  ------  -------  -----  ------
0000  0400  PRDV01     P   33901        0.0      0.0      0.0    0.0     0.0    0.0    0.0    0.00      0.0    0.0     0.0
0001  0401  PRDV02     P   33901        0.4      0.2      0.7    2.4    70.8   98.8   79.1    0.00      0.1    3.8    38.9
0002  0402  PRDV03     P   33901        0.1      0.0      0.1    5.9    78.5  100.0   81.6    0.00      0.0    4.4     9.1
0003  0403  PRDV04     P   33901        0.4      0.3      0.8    1.1    74.5   95.3   84.4    0.00      0.1    5.4    49.6
0004  0404  PRDV05     P   33901        0.2      0.1      0.3    1.4    62.5  100.0   78.0    0.00      0.1    3.4    28.8
0005  0405  PRDV06     P   33901        0.0      0.0      0.0    5.0    36.7  100.0   47.2    0.00      0.0    0.7    52.6
0006  0406  PRDV07     P   33901        0.1      0.0      0.1    3.9    68.3  100.0   74.8    0.00      0.0    2.9    10.3
0007  0407  PRDV08     P   33901        0.0      0.0      0.1    1.7    63.6   94.7   75.0    0.00      0.0    3.0     0.4
0008  0408  PRDV09     P   33902        0.0      0.0      0.1    2.1    81.8  100.0   87.8    0.00      0.0    5.4    23.5
0009  0409  PRDV10     P   33902        5.9      0.0      6.1   17.6    91.1  100.0   91.1    0.00      0.6    8.7     1.9
000A  040A  PRDV11     P   33902        0.4      0.7      1.1    0.5    83.6   97.6   93.1    0.00      0.1   12.7    13.9
000B  040B  PRDV12     P   33902        0.1      0.0      0.1    2.5    87.2   97.1   90.1    0.00      0.0    6.8    52.9
000C  040C  PRDV13     P   33902        2.4      0.0      2.6   50.7    90.0   97.7   90.1    0.00      0.2    9.1     3.8
000D  040D  PRDV14     P   33902        1.2      0.1      1.4    8.6    86.5  100.0   87.9    0.00      0.2    7.3    31.7
000E  040E  PRDV15     P   33902        0.0      0.0      0.0    3.7    59.1  100.0   67.9    0.00      0.0    2.1    11.1
000F  040F  PRDV16     P   33902        0.1      0.1      0.2    1.7    90.1  100.0   93.8    0.00      0.0   13.5    42.9
0010  0410  PRDV17     P   33902        1.0      0.2      1.3    6.3    79.3   98.6   81.9    0.00      0.2    4.5    14.8
⋮
00FD  04FD  ORVM14     P   33902        2.1      0.0      2.0   87.9    90.9   95.2   90.9    0.00      0.2    7.6     9.5
00FE  04FE  ORVM15     P   33902        0.3      0.0      0.2    0.0    73.2    0.0   73.2    0.00      0.1    2.7    78.7
00FF  04FF  ORVM16     P   33902        0.1      1.3      1.4    0.1    88.0   70.9   71.9    0.00      0.4    2.6    32.4

================================================================================================================================

     SUBSYSTEM              READ    WRITE     I/O    READ    READ   WRITE   I/O    DFW     STAGE   HITS/    LOW   TRACK
      SUMMARY             PER SEC  PER SEC  PER SEC  RATIO   HIT %  HIT %  HIT %  CONSTR  PER SEC   STGE  REF CT  OCCUP
                          -------  -------  -------  -----   -----  -----  -----  ------  -------  -----  ------  ------
     PROD PARTITION         320.7    110.6    436.1    2.9    87.3   97.6   89.9     0.0     50.6    7.7    35.7
     OVERALL TOTALS         320.7    110.6    436.1    2.9    87.3   97.6   89.9     0.0     50.6    7.7    35.7   218.0
================================================================================================================================

 REPORT INCLUSION DAYS: ALL
 REPORT EXCLUSION DAYS: N/A
 REPORT EXCLUSION DATES: N/A
```

Notice the subsystem summary at the end of the device report. Also, the **TRACK OCCUP** column appears only in the overall totals line of the subsystem summary, not in the device-detail reports.

Cache Effectiveness Overall Summary

The overall summary provides averages over the whole range covered by the report. The summary has the same format as the report for the individual intervals.

```
REPORTER                                CACHE EFFECTIVENESS OVERALL SUMMARY                 16NOV1999    15:22:53
                                                                                           SIBCEIO  V1 R0 L1
REPORT START DATE: 10OCT1999      REPORT END DATE: 10OCT1999
REPORT START TIME:  8:14          REPORT END TIME: 11:59

SUBSYSTEM NAME: STK         (CACHE SIZE: 512 MB    NVS SIZE:  8 MB)

FDID  DEV   VOLSER    T/P  DEVICE     READ    WRITE     I/O    READ    READ   WRITE   I/O    DFW    STAGE    HITS/    LOW
      ADDR                TYPE     PER SEC  PER SEC  PER SEC  RATIO   HIT %  HIT %  HIT %  CONSTR  PER SEC   STGE  REF CT
----  ----  --------  ---  ------   -------  -------  -------  -----   -----  -----  -----  ------  -------  -----  ------
0000  0400  PRDV01     P   33901        0.0      0.0      0.0    0.0    72.7    0.0   72.7    0.00      0.0    2.7     0.0
0001  0401  PRDV02     P   33901        1.2      0.2      1.4    5.8    88.8   98.6   90.2    0.00      0.2    7.1    55.2
0002  0402  PRDV03     P   33901        0.4      0.3      0.6    1.4    93.0   99.5   95.8    0.00      0.0   20.2    19.1
0003  0403  PRDV04     P   33901        2.7      0.2      3.0   11.1    50.1   94.9   53.8    0.00      1.4    1.2    17.3
0004  0404  PRDV05     P   33901        0.1      0.1      0.1    1.1    54.2  100.0   76.3    0.00      0.0    3.1    25.6
0005  0405  PRDV06     P   33901        0.1      0.0      0.1   12.5    76.0  100.0   77.8    0.00      0.0    2.7    38.3
0006  0406  PRDV07     P   33901        0.1      0.0      0.2    4.7    87.0   86.7   87.0    0.00      0.0    5.3    33.4
0007  0407  PRDV08     P   33901        0.8      0.0      0.8   53.7    62.4   99.5   63.1    0.00      0.3    1.7     2.9
0008  0408  PRDV09     P   33902        0.1      0.0      0.1    9.8    89.8  100.0   90.1    0.00      0.0    2.7    73.3
0009  0409  PRDV10     P   33902        1.1      0.0      1.1  100.6    73.6   99.3   73.9    0.00      0.3    2.7     2.4
000A  040A  PRDV11     P   33902        0.2      0.3      0.5    0.5    82.2   97.9   92.7    0.00      0.0   10.9    28.7
000B  040B  PRDV12     P   33902        0.7      0.0      0.8   62.3    54.3   96.8   55.0    0.00      0.4    1.1    10.2
000C  040C  PRDV13     P   33902        1.4      0.0      1.6   36.0    83.6   98.9   84.0    0.00      0.2    5.2     5.7
000D  040D  PRDV14     P   33902        1.4      0.2      1.7    5.9    88.0   99.9   89.7    0.00      0.2    6.2    26.9
000E  040E  PRDV15     P   33902        2.2      0.0      2.3  213.2    57.0   97.8   57.2    0.00      0.9    1.3     0.8
000F  040F  PRDV16     P   33902        1.3      0.0      1.4  118.6    52.4   99.3   52.8    0.00      0.6    1.1     2.0
0010  0410  PRDV17     P   33902        0.8      0.1      0.9    6.7    80.6   98.5   83.0    0.00      0.2    4.6    29.8
⋮
00FD  04FD  ORVM14     P   33902        1.0      0.1      0.9   16.4    94.3   95.0   94.3    0.00      0.1    9.5    12.8
00FE  04FE  ORVM15     P   33902        0.8      0.0      0.6  190.4    94.1   83.6   94.0    0.00      0.1   11.6    25.4
00FF  04FF  ORVM16     P   33902        0.2      1.8      2.1    0.1    97.4   91.1   91.8    0.00      0.2    9.7    14.0

================================================================================================================================

     SUBSYSTEM              READ    WRITE     I/O    READ    READ   WRITE   I/O    DFW     STAGE   HITS/    LOW   TRACK
      SUMMARY             PER SEC  PER SEC  PER SEC  RATIO   HIT %  HIT %  HIT %  CONSTR  PER SEC   STGE  REF CT  OCCUP
                          -------  -------  -------  -----   -----  -----  -----  ------  -------  -----  ------  ------
     PROD PARTITION         253.3     94.9    353.6    2.7    86.2   98.5   89.6     0.0     41.7    7.5    30.8
     OVERALL TOTALS         253.3     94.9    353.6    2.7    86.2   98.5   89.6     0.0     41.7    7.5    30.8   205.9
================================================================================================================================

  REPORT INCLUSION DAYS: ALL
  REPORT EXCLUSION DAYS: N/A
  REPORT EXCLUSION DATES: N/A
```

Reading Space Utilization Reports

This section describes the layout of the default space utilization reports. See “Definitions of Field Names” on page 9-15 for an explanation of the field names.

Sample Space Utilization Report

The listing below shows a space utilization report.

The following subcommand produced the report.

```
REPORT SPACEU
```

```
REPORTER                                SPACE UTILIZATION SUMMARY REPORT                    18NOV1999     6:26:02
                                                                                           SIBSPUT  V2 R1 L1

SUBSYSTEM INT                    (NUMBER OF FUNCTIONAL DEVICES: 737)

                                                --FUNCTIONAL CAPACITY (MB)--  % FUNCTIONAL CAPACITY
FDID  DEV   VOLSER    T/P  DEVICE  FUNCT                          NOT                   NOT   --PHYSICAL CAP USED (MB)--   COMP
      ADDR                 TYPE   CAP (MB)    ALLOC    STORED    STORED    ALLOC STORED STORED   SHARED   UNIQUE   TOTAL     RATIO
----  ----  ------    ---  ------  --------  --------  --------  --------  ----- ------ ------  -------- -------- --------   -----
⋮
000A  N/A   INT010     P   33901      946.0       N/A       3.5     942.5     N/A    0.4   99.6       0.0      0.3      0.3    11.2
000B  N/A   INT011     P   33901      946.0       N/A       3.5     942.5     N/A    0.4   99.6       0.0      0.3      0.3    11.2
000C  N/A   INT012     P   33901      946.0       N/A       3.5     942.5     N/A    0.4   99.6       0.0      0.3      0.3    11.2
⋮
0052  N/A   INT082     P   33902     1892.0       N/A      27.5    1864.5     N/A    1.5   98.5       0.0     14.1     14.1     1.9
0053  N/A   N/A    *   P   33902     1892.0       N/A      33.9    1858.1     N/A    1.8   98.2       0.0     18.4     18.4     1.8
0054  N/A   INT084 *   P   33902     1892.0       N/A      23.4    1868.6     N/A    1.2   98.8       0.0     10.0     10.0     2.3
0055  N/A   INT085     P   33902     1892.0       N/A      55.3    1836.7     N/A    2.9   97.1       3.2     10.7     13.9     4.0
⋮
0080  N/A   AAA0080    P   SCSIA     1230.2       N/A       0.1    1230.1     N/A    0.0  100.0       0.0      0.0      0.0*    7.2
0081  N/A   AAA0081    P   SCSIA     1230.2       N/A       0.1    1230.1     N/A    0.0  100.0       0.0      0.0      0.0*    7.2
0082  N/A   F82        P   SCSIA     1230.2       N/A    1223.4       6.8     N/A   99.5    0.5     889.4      0.2    889.6     1.4
0083  N/A   F83        P   SCSIA     1230.2       N/A    1223.4       6.8     N/A   99.5    0.5     889.4      0.2    880.6     1.4
0084  N/A   F261       P   SCSIA     1230.2       N/A     268.4     961.8     N/A   21.8   78.2       0.0    186.1    186.1     1.4
0085  N/A   F1100      P   SCSIA     1230.2       N/A       0.1    1230.0     N/A    0.0  100.0       0.0      0.0      0.0*    8.6
0086  N/A   COSMO67    P   SCSIB     3691.9       N/A       0.1    3691.8     N/A    0.0  100.0       0.0      0.0      0.0*    7.5
0087  N/A   COSMO68    P   SCSIB     3691.9       N/A       0.1    3691.9     N/A    0.0  100.0       0.0      0.0      0.0*    6.0
⋮
00F7  N/A   INT247     P   3380KE    1260.5       N/A    1259.8       0.7     N/A   99.9    0.1       0.0    336.5    336.5     3.7
00F8  N/A   INT248     P   3380KE    1260.5       N/A       1.1    1259.4     N/A    0.1   99.9       0.1      0.0      0.1    11.3
00F9  N/A   INT249     P   3380KE    1260.5       N/A       1.2    1259.3     N/A    0.1   99.9       0.1      0.0      0.1    11.8
00FA  N/A   INT250     P   3380K     1890.7       N/A    1890.1       0.7     N/A  100.0    0.0       0.0    504.8    504.8     3.7
00FB  N/A   INT251     P   3380K     1890.7       N/A    1890.1       0.7     N/A  100.0    0.0       0.0    504.8    504.8     3.7
00FC  N/A   INT252     P   3380K     1890.7       N/A       0.2    1890.5     N/A    0.0  100.0       0.0      0.0      0.0*    9.3
00FD  N/A   INT253     P   3380K     1890.7       N/A       0.2    1890.5     N/A    0.0  100.0       0.0      0.0      0.0*    9.3
00FE  N/A   INT254     P   3380K     1890.7       N/A       0.2    1890.5     N/A    0.0  100.0       0.0      0.0      0.0*    9.3
00FF  04FF  INT255     P   3380K     1890.7       0.2       0.2    1890.5     0.0    0.0  100.0       0.0      0.0      0.0*    9.3

SELECTED DEVICES SUMMARY                      FUNCTIONAL CAPACITY (MB)    % FUNCT CAPACITY   -------- DISK ARRAY --------
                    SELECTED  TOTAL FUNCTIONAL                  NOT                 NOT       -- PHYSICAL CAP USED (MB) --  COMP
                     DEVICES  CAPACITY (MB)          STORED    STORED       STORED STORED      SHARED     UNIQUE    TOTAL   RATIO
                    --------  -------------       --------- ---------      ------ ------     --------  --------  --------  -----
PRODUCTION PARTITION:     576     1619708.3           1286.1 1618422.2          0.1   99.9          0.0     166.8     166.9    7.7
TOTALS:                   576     1619708.3           1286.1 1618422.2          0.1   99.9          0.0     166.8     166.9    7.7

SELECTED SCSI DEVICES SUMMARY                 FUNCTIONAL CAPACITY (MB)    % FUNCT CAPACITY   -------- DISK ARRAY --------
                    SELECTED  TOTAL FUNCTIONAL                  NOT                 NOT       -- PHYSICAL CAP USED (MB) --  COMP
                     DEVICES  CAPACITY (MB)          STORED    STORED       STORED STORED      SHARED     UNIQUE    TOTAL   RATIO
                    --------  -------------       --------- ---------      ------ ------     --------  --------  --------  -----
PRODUCTION PARTITION:     161      371505.0            6292.6  365212.4          1.7   98.3       2725.7     324.5    3050.2    2.1
TOTALS:                   161      371505.0            6292.6  365212.4          1.7   98.3       2725.7     324.5    3050.2    2.1

SUBSYSTEM INT              SPACE UTILIZATION SUMMARY
                     NUMBER OF          DISK ARRAY      NET CAPACITY LOAD(%)     COLL FREE SPACE (%)      UNCOLL FREE SPACE(%)
                FUNCTIONAL DEVICES     CAPACITY (MB)     TEST   PROD  OVERALL     TEST   PROD  OVERALL     TEST   PROD  OVERALL
                ------------------     -------------    -----  -----  -------    -----  -----  -------    -----  -----  -------
                               737          113346.4       0.0    2.6     2.6       0.0   97.1    97.1       0.0    0.3     0.3
```

Notes:

1. A “PHYS CAP USED--TOTAL” field containing 0.0* indicates that a small amount of back end storage exists.
2. In reports:
   - “N/A” indicates that the device address is not available or invalid.
   - An asterisk to the right of an entry in the VOLSER column indicates that the device is offline to the host.
   - V-ERR in an ALLOC column indicates that errors have occurred in accessing the VTOC. Look for messages in the job log with information about the error.

Bar Charts

A bar-chart report shows values you specify from one or more of the other seven types of report, plotted over time on a bar chart (histogram). The sample reports below show bar charts for the access density, drive module coefficient of variation, and I/O rate. The statistics were collected at 1-minute intervals and summarized at 30-minute intervals.

This REPORT PERF subcommand produced the report shown on the following pages:

```
REPORT PERF-
    (DEVPERF (YES)-
    BAR (ACCDEN DRVMCOV IORTE) )
```

A bar chart report has these parts:

1. The common report header.
2. The SVAA subsystem name and the variable shown on the bar graph.
3. The bar graph.

```
   REPORTER                                   BAR CHART REPORT                                 16NOV1998   9:53:45
                                                                                                SIBCHART V1 R0 L1

---------------------------------------------- SUBSYSTEM=STK VARIABLE=Access Density ----------------------------------------

                            HIGHTD   PARTMEMB

                                              |
                       21OCT98:08:44   Total   |***********************
                                              |
                       21OCT98:09:14   Total   |***********************
                                              |
                       21OCT98:09:44   Total   |*************************
                                              |
                       21OCT98:10:14   Total   |****************************
                                              |
                                              |
                                              ----------+---------+---------+---------+---------+--------
                                                        1         2         3         4         5

                                                              SUM OF VARIABLE BY INTERVAL
```

```
REPORTER                                    BAR CHART REPORT                        16NOV1998        9:53:45

------------------------------------ SUBSYSTEM=STK VARIABLE=Drive Module Coeff of Var ------------------------------------

                                    |
21OCT98:17:14   Total               |*******************
                                    |
21OCT98:17:44   Total               |*******************
                                    |
21OCT98:18:14   Total               |************************
                                    |
21OCT98:18:44   Total               |************
                                    |
21OCT98:19:14   Total               |******************************
                                    |
21OCT98:19:44   Total               |***********************************
                                    |
21OCT98:20:14   Total               |************************************************
                                    |
21OCT98:20:44   Total               |****************************************
                                    |
21OCT98:21:14   Total               |*********************************************
                                    |
21OCT98:21:44   Total               |****************************************
                                    |
21OCT98:22:14   Total               |***********************************
                                    |
21OCT98:22:44   Total               |*******************************
                                    |
21OCT98:23:14   Total               |******************************
                                    |
21OCT98:23:44   Total               |****************************************
                                    |
21OCT98:23:59   Total               |*****************************************
                                    |
22OCT98:00:00   Total               |**************************************
                                    |
22OCT98:01:14   Total               |*********************************************
                                    |
                                    ----------+---------+---------+---------+---------+--------
                                              1         2         3         4         5
```

```
REPORTER                                                BAR CHART REPORT          16NOV1998     7:08:30       7
                                                                                  SIBCHART V1 R0 L0

------------------------------------------- SUBSYSTEM=STK VARIABLE=IORTE ---------------------------------------------

                    HIGHTD   PARTMEMB

                                    |
              21OCT98:13:50 Production|*****************************************************************
                            Test    |*****************************************************************
                            Total   |*****************************************************************
                                    |
                                    ----------+---------+---------+---------+---------+---------+--------
                                              1         2         3         4         5         6

                                                    SUM OF VARIABLE BY INTERVAL
```

Graphic Reports

Users with SAS/GRAPH® installed can request graphic displays of interval, time-ordered, daily, weekly, or monthly summary report data. Reporter saves the summarized data from the report in SAS data set format to be sent to a separate SAS/GRAPH program, which you must run later. The SVAA SIBSAMP data set includes sample SAS/GRAPH programs for producing graphics output from the saved data. The input data to SAS/GRAPH will already have been summarized; no further SAS computation is necessary. SAS/GRAPH output includes trend analyses.

Definitions of Field Names

The following list defines the field names and column headings of the reports explained in this manual. (“Variables Used in Definitions” on page 9-21 explains the variables in the formulas shown in the list.)

% ACTIVE ON CHNL
The percentage of the reporting interval that the subsystem was active on this channel interface. The time that the subsystem was active on the channel interface is the total of all the connect times on the interface.

% COLLECTED FREE SPACE
The percentage of the array cylinders in the subsystem that are free array cylinders (that is, the total space that can be written to).

% DEV AVAIL The percentage of the reporting interval in which the device was available (operational).

$$= \frac{\text{device available time} * 100}{\text{durms}}$$

% DEV CONN The percentage of the reporting interval in which the device was connected to the channel while processing an I/O operation.

$$= \frac{\text{conntime} * 100}{\text{durms}}$$

Note: For FICON channels, this percentage may exceed 100%.

% DEV DISC The percentage of the reporting interval in which the device was disconnected from the channel while processing an I/O operation.

$$= \frac{\text{devutlti-conntime} * 100}{\text{durms}}$$

% DEV UTIL The percentage of the reporting interval in which the device was utilized (busy). Busy time = connect time + disconnect time.

$$= \frac{\text{devutlti} * 100}{\text{durms}}$$

% DRIVE MODULE UTILIZATION
The highest percentage of drive module utilization in each decile range. This is a measure of the evenness of drive module utilization in the subsystem. A perfectly even distribution of drive module utilization would contain the same value in each percentile.

% FUNCTIONAL CAPACITY - NOT STORED
The percentage of defined functional capacity that is not occupying back-end storage.

% FUNCTIONAL CAPACITY - STORED
The percentage of defined functional capacity that is occupying back-end storage.

% FUNCTIONAL CAPACITY - ALLOC
The percentage of defined functional capacity that is allocated (in the VTOC).

Note: For SCSI volumes, the calculated percentage may exceed 100% due to rounding and how the functional capacity of SCSI volumes is calculated.

ACCESS DENSITY
The number of I/O operations per second per gigabyte of functional capacity defined.

$$= \frac{\text{iorate}}{\text{functional capacity in GB}}$$

AVG % DRIVE MODULE UTIL
The average percentage of time that the drive module was busy for the reporting interval, weighted by interval duration.

Definitions of Field Names

CACHE SIZE The size in megabytes of cache storage available to the user.

CHANNEL SPEED
The transfer rate (in megabytes per second) of the channel interface.

CLUSTER The channel interface cluster identifier. (Values: 0 or 1.)

COEFF OF VARIATION
The coefficient of variation of the percentages of times the drive modules were busy. This is a measure of the evenness of utilization of the drive modules in the subsystem; the smaller the coefficient of variation, the more evenly distributed is the utilization of the drive modules (activity is more evenly spread among the drive modules). Ideally, the utilization of each drive module is equal to the mean of all drive-module utilizations and the coefficient of variation is zero. A coefficient of variation greater than 20 should be investigated.

COLL FREE SPC
The space in the free array cylinders in the subsystem (that is, the total space that can be written to).

COLL FREE SPACE %–OVERALL
The percentage of the array cylinders in the subsystem that are free array cylinders.

$$= \frac{\text{frebescp + frebesct}}{\text{totbecpp + totbecpt}} * 100$$

COLL FREE SPACE %–PROD
The percentage of the array cylinders in the Production partition that are free array cylinders.

$$= \frac{\text{frebescp}}{\text{totbecpp}} * 100$$

COLL FREE SPACE %–TEST
The percentage of the array cylinders in the Test partition that are free array cylinders.

$$= \frac{\text{frebesct}}{\text{totbecpt}} * 100$$

COMP RATIO The approximate ratio of functional capacity stored to the physical capacity used (on a device level).

$$= \frac{\text{capstored}}{\text{phys cap used}}$$

DATE The ending date on which data included in this report was collected.

DEV ADDR The address of the device as defined in the IOCP.

DEVICE TYPE The type of DASD of the functional device. (Values: 3380-J, 3380-K, 3390-KE, 3390-1, 3390-2, 3390-3, 3390-9.)

DFW CONSTR The number for a functional device or the percentage for a subsystem of DASD fast-write operations that were bypassed (the data was written directly to DASD because insufficient nonvolatile storage was available). The value should be zero for SVA subsystems.

For devices:

= NVS constraint count

For subsystems:

$$= \frac{\text{NVS constraint count}}{\text{fastwrts}} * 100$$

DISK ARRAY CAPACITY
The amount of back-end storage available in the subsystem.

$$= \frac{\text{totbecpp + totbecpt}}{1000000}$$

DISTRIBUTION OF DRIVE MODULE UTILIZATION
A table showing at intervals of 10 percent the percentages of drive-module utilization.

DRIVE MODULE UTIL COEFF OF VAR
Coefficient of variation of individual drive module busy times.

EXCEPTION THRESHOLDS
The exception values that were specified for certain statistics. Values in the report that exceed the exception thresholds are marked with an *.

FDID The SVA subsystem identifier for the functional device. (Values: 00 - FF.)

FREE SPACE COLLECTION LOAD
The amount of back-end physical space collected for free-space consolidation that did not yield available free space.

$$= \frac{\text{bytes read for free-space collection}}{\text{bytes of free space collected}}$$

FREQUENCY The percentiles at which drive module utilization is reported.

FUNCT CAP The functional capacity in megabytes defined for the device.
For CKD devices:

$$= \frac{\text{bytes per track} * \text{tracks per cyl} * \text{primcapa}}{1000000}$$

For SCSI devices:

$$= \frac{\text{scsilbt} * \text{scsilbsz} * \text{tracks per cyl} * (\text{primcapa} - 2)}{1000000}$$

FUNCTIONAL CAPACITY–ALLOCATED
The functional space read from the VTOC.

$$= \frac{\text{total allocated functional space}}{1000000}$$

Note: For SCSI volumes, the allocated functional capacity is the sum of the SCSI device's functional capacity plus the first two cylinders of the volume. As a result, the values reported will be greater than the functional capacity of a SCSI volume.

FUNCTIONAL CAPACITY–NOT STORED
The part of the functional capacity defined for the device or subsystem that is not mapped (that is, is not occupying back-end storage).
For CKD devices:

$$= \frac{\text{functional capacity} - \text{functional capacity stored}}{1000000}$$

FUNCTIONAL CAPACITY–STORED
The functional capacity in megabytes mapped (occupying back-end storage) for the device or subsystem.
For CKD devices:

$$= \frac{\text{bytes per track} * \text{number of tracks mapped}}{1000000}$$

For SCSI devices:

$$= \frac{\text{scsilbt} * \text{scsilbsz} * \text{number of tracks mapped}}{1000000}$$

HITS/STGE The ratio of the number of channel segments for which the addressed track was in cache storage to the number of times data was transferred from DASD storage to cache storage (that is, to the number of times the addressed track was not in cache storage). This is the ratio of cache hits to cache misses.

$$= \frac{\text{readhits + writhits}}{\text{stages}}$$

I/O HIT % The percentage of all I/O operations (read or write) for which the addressed track was present in cache storage.

$$= \frac{\text{readhits + writhits}}{\text{readreqs + writreqs - offlnrds}} * 100$$

I/O PER SEC The average number of I/O operations per second for the device or subsystem.

$$= \frac{\text{devactiv}}{\text{dursecs}}$$

Note: In Cache Effectiveness reports, I/O PER SEC may not be equal to the sum of READ PER SEC and WRITE PER SEC, either because *devactiv* may include other I/O operations, such as sense commands, or because there may be more than one read or write operation per channel program.

I/O SERVICE TIME–CONNECT
The average time the device was connected to the channel (in milliseconds) while processing an I/O operation.

$$= \frac{\text{conntime}}{\text{devactiv}}$$

This includes data transfer time and command parameter transfer time.

I/O SERVICE TIME–DISC
The average time the device was disconnected from the channel (in milliseconds) while processing an I/O operation.

$$= \frac{\text{devutlti - conntime}}{\text{devactiv}}$$

I/O SERVICE TIME–TOTAL
The average service time per I/O operation in milliseconds.

$$= \frac{\text{devutlti}}{\text{devactiv}}$$

The time does not include host queuing time (that is, IOSQ and Pend time.)

INTERFACE ID
The channel interface identifier. (Values: One of A through P.)

INTERFACE NAME
The name you assigned to the channel interface.

INTERVAL DURATION
The length of the collection interval in hours and minutes.

KBYTES PER SECOND
The amount of data in kilobytes transferred per second between the host and the subsystem.

$$= \frac{\text{febytxfr + febytxfw}}{\text{1000 * dursecs}}$$

LOW REF CT The average number of deallocated frames in cache that had fewer than two references before deallocation.

$$= \frac{\text{number of frames deallocated with fewer than 2 references}}{\text{frame deallocation count}} * 100$$

NET CAPACITY LOAD
The amount of back-end physical capacity used in a subsystem. This includes user data and the system areas needed to maintain the arrays. NCL does not include data in the cache until the data is written to the back end.

NET CAPACITY LOAD %
The percentage of back-end physical capacity that is used (not free).

$$= \frac{\text{total back-end capacity - free back-end capacity}}{\text{total back-end capacity}} * 100$$

Note: The percentage can exceed 100 if the Capacity on Demand feature is installed and PCAP is exceeded.

NOMINAL FUNCTIONAL CAPACITY
The total defined capacity of all the devices. This is the functional capacity that would be available if every track of every device had a single record with a block length equal to the track size.
= functional capacity stored + funct cap not stored

NUMBER OF FUNCTIONAL DEVICES
The number of functional devices defined with back-end storage in the subsystem.

NVS SIZE The size in megabytes of the nonvolatile storage.

OFFLINE CACHE
The amount of cache storage that has been made unavailable.

OVERALL TOTALS
The sum of the Test partition and Production partition totals. For statistics that are ratios, including percentages, the totals are the weighted averages of the individual statistics.

PHYSICAL CAP USED (MB) - SHARED
The amount of back-end physical space (in megabytes) that is used (shared) by multiple functional devices as a result of a snap command.

PHYSICAL CAP USED (MB) - UNIQUE
The amount of back-end physical space (in megabytes) that is used by only one functional device.

PHYSICAL CAP USED (MB) - TOTAL
The total (shared + unique) back-end physical space (in megabytes) that is used by a functional device.

Note: The values reported for the three terms above do not include the storage used by the mapping tables and functional track directories used to maintain the disk arrays.

PINNED CACHE
The amount of cache that is unavailable to the subsystem because of cache failures or inability to destage a track from the cache. In SVA reports, this field should always be 0.

PROD PARTITION or PRODUCTION PARTITION
The statistics totals for devices in the Production partition. For statistics that are ratios, including percentages, the totals are weighted averages of the individual statistics.

READ HIT % The percentage of read operations for which the addressed track was present in cache storage.

$$= \frac{\text{readhits}}{\text{readreqs - offlnrds}} * 100$$

READ PER SEC
The average number of read operations per second for the device.

$$= \frac{\text{readreqs}}{\text{dursecs}}$$

READ/WRITE RATIO
The ratio of the number of read operations to write operations.

$$= \frac{\text{readreqs}}{\text{writreqs}}$$

REPORT END DATE
The latest ending date found in records in the file within the selected date range.

REPORT END TIME
The latest ending time found in records in the file within the selected date range and time range.

REPORT EXCLUSION DATES
The dates explicitly excluded from the report.

REPORT EXCLUSION DAYS
The days of week excluded from the report.

REPORT INCLUSION DAYS
The days of week included in the report.

REPORT START DATE
The earliest beginning date found in records in the file within the selected date range.

REPORT START TIME
The earliest beginning time found in records in the file within the selected date range and time range.

SIBCExx The name of the SAS macro that produced the Cache Effectiveness reports. This is followed by the SVAA version, release, and level.

SIBDPxx The name of the SAS macro that produced the Device Performance reports. This is followed by the SVAA version, release, and level.

STAGE PER SEC
The average number of staging operations (transfers of data from DASD storage to cache storage) per second.

$$= \frac{\text{stages}}{\text{dursecs}}$$

SUBSYSTEM The name of the SVA subsystem to which this report applies.

SUBSYSTEM NAME
The name of the SVA subsystem to which this report applies.

SUMMARY REPORTS
One of the Daily, Weekly, or Monthly Summary reports provided by Reporter.

SYNCHRONIZATION
In data collection, the number of minutes after the hour on which data-collection intervals are to begin.

T/P The partition to which the functional device belongs at the end of the reporting interval. (Values: T or P.)

TEST PARTITION
The statistics totals for devices in the Test partition. For statistics that are ratios, including percentages, the totals are weighted averages of the individual statistics.

TIME The time that the data was collected from the subsystem (in SVAA and RMF reports, this shows the *ending* time of the interval; in MICS reports, it is the *starting* time).

TOTAL FUNCTIONAL CAPACITY
The total defined functional capacity of all the devices, in megabytes. This is the functional capacity that would be available if every track of every device had a single record with a block length equal to the track size.

= functional capacity stored + functional capacity not stored

TOTAL FUNCTIONAL CAPACITY STORED
The total number of megabytes of data stored on DASD.

$$= \frac{\text{sum of capacity stored for all volumes}}{1000000}$$

TOTALS A field that shows the Production partition and Test partition totals. For statistics that are ratios, including percentages, the totals are averages of the individual statistics.

TRACK OCCUP The average number of tracks per megabyte of cache per second. (This field does not appear on detail reports.)

$$= \frac{\text{cache track seconds}}{\text{dursecs} * \text{MB of cache}}$$

UNCOLL FREE SPACE %
The original space occupied when a functional track has been rewritten to a new location in the disk array. Also, the original space occupied by deleted tracks reflected in the VTOC when a data set or volume has been deleted and DDSR has been run.

VOLSER The volume serial number for the device at the end of the reporting interval.

If **N/A** appears in the DEV ADD column, the functional device is not defined to the host operating system. In that case, **N/A** also appears in the VOLSER column.

WRITE HIT % The percentage of all write operations for which the addressed track was present in cache storage.

$$= \frac{\text{writhits} * 100}{\text{fastwrts}}$$

WRITE PER SEC
The average number of write operations per second for the device.

$$= \frac{\text{writreqs}}{\text{dursecs}}$$

Variables Used in Definitions

The formulas on the preceding pages may include the following variable names:

capstord Functional capacity stored

CFW Cache fast write

conntime Total device connection time in milliseconds

devactiv Number of end-of-chain events (that is, the number of I/Os) for a device

devutlti Time in milliseconds the device is utilized (busy)

DFW DASD fast write

Definitions of Variables

durms Duration of the interval in milliseconds

dursecs Duration of the interval in seconds

falloc Functional capacity allocated:
total allocated functional space (from the VTOC) / 1000000

fastwrts Number of DFW normal requests + DFW sequential requests + CFW requests

febytxfr Number of front-end read bytes transferred

febytxfw Number of front-end write bytes transferred

frebecpp Total (collected and uncollected) free space for Production partition

frebecpt Total (collected and uncollected) free space for Test partition

frebescp Collected free space for Production partition

frebesct Collected free space for Test partition

iorate devactiv/dursecs

ncapld Net capacity load

numberio Number of end-of-chain events at the channel interface

offlnrds Number of reads that occurred while caching was deactivated for a device

primcapa Primary capacity in cylinders

readhits Number of read normal hits + read sequential hits + CFW hits

readreqs Number of read normal requests + read sequential requests + CFW requests

scsilbsz SCSI logical block size

scsilbt SCSI logical blocks per track

stages Number of DASD-to-cache transfers

stndrd Standard functional capacity

stored Total functional capacity stored

totbecpp Total back-end capacity (Production partition)

totbecpt Total back-end capacity (Test partition)

writreqs Number of write normal requests + write sequential requests + CFW requests

writhits Number of DFW normal hits + DFW sequential hits + CFW hits

xtnd Extended functional capacity

Chapter 10. Format of Logging Records

Chapter Summary

The Data Collection File

This chapter shows the layout of the records written in the Reporter data collection file. Output can be directed either to SMF or to a user-managed file. In each case, a standard SMF record header precedes the Reporter data portion of the record, although it is mainly binary zeroes if output is not directed to SMF.

The five Reporter data collection record subtypes are:

- Subtype 1: Subsystem Performance
- Subtype 2: Channel Interface Statistics
- Subtype 3: Functional Device Performance
- Subtype 4: Drive Module Performance
- Subtype 7: Space Utilization

Two other SVAA SMF record subtypes are documented in the appendix titled "SMF Record Formats" in the *SVAA for OS/390 Configuration and Administration* manual:

- Subtype 5: DDSR data
- Subtype 8: SnapShot event data

Data collection record subtypes, as shown in Figure 10-1, consist of:

1. The SMF record header (24 bytes)
2. The Data Collection record prefix (56 bytes)—not present in the Space Utilization record
3. Information Segment Descriptors (8 bytes each)
4. Information Segments (variable length)

SMF Record Header

The SMF record prefix provides general information about the record itself, such as its length, and date and time written. A two-byte field at offset 22 identifies the record subtype. The first part of Table 10-1 on page 10-3 shows the format of the SMF record header.

Data Collection Record Prefix

The Data Collection record prefix provides general information about the conditions under which statistics are reported. The record prefix is present only in record subtypes 1 through 4. The second part of Table 10-1 on page 10-3 shows the format of the record prefix.

Information Segment Descriptors

There is one 8-byte Information Segment Descriptor area for each Information Segment that follows. Data in the segment descriptor includes the offset to the segments, the size of the segments, and the number of them.

If no Information Segments follow, the Information Segment Descriptors are set to zeros.

Information Segments

Information Segments contain the statistics gathered from the subsystems. Information Segments vary in size; they can be omitted, or more than one can be present, depending on the type of record.

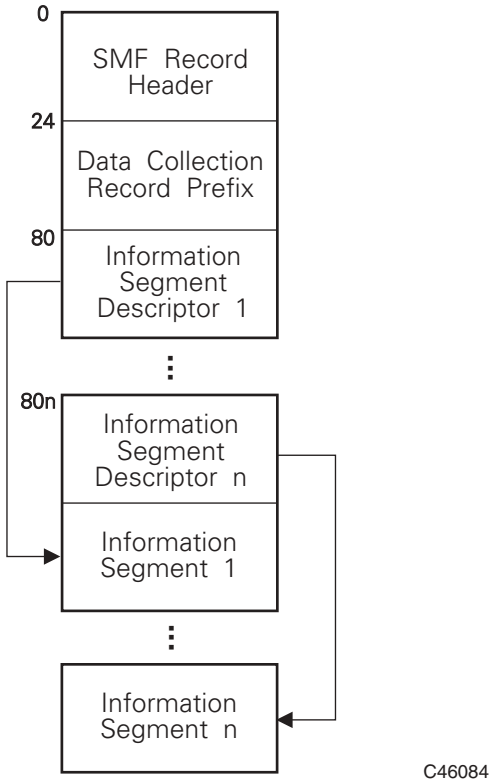


Figure 10-1. *General format of Data Collection records, subtypes 1-4*

Table 10-1 (Page 1 of 2). *Record header and Data Collection record prefix for subtypes 1-4*

Dec Offset	Hex Offset	Field Name	Length	Type	Description
			SMF Record Header		
0	00	len	2	Binary	Record length. This field and the next make up the record descriptor word.
2	02	seg	2	Binary	Segment descriptor
4	04	smfflg	1	Binary	X'80' New SMF record format X'40' Subtype utilized X'20' (reserved) X'10' MVS/SP Version 4 X'08' MVS/ESA X'04' MVS/XA X'02' OS/VS2 X'01' (reserved)
5	05	smfrty	1	Binary	SMF record number
6	06	smftime	4	Binary	Time record written, in hundredths of seconds since midnight
10	0A	smfdate	4	Packed	Date record written, in the form 0*cyydddF* where: c=0 for 1990 to 1999 c=1 for 2000 to 2099
14	0E	smfsid	4	Character	System ID
18	12	smfssi	4	Character	Subsystem identification for SMF

Table 10-1 (Page 2 of 2). *Record header and Data Collection record prefix for subtypes 1-4*

Dec Offset	Hex Offset	Field Name	Length	Type	Description
22	16	rectype	2	Binary	SMF record subtype X'0001' Subsystem Performance X'0002' Channel Interface Statistics X'0003' Functional Device Performance X'0004' Drive Module Performance X'0005' DDSR X'0006' (reserved) X'0007' Space Utilization X'0008' SnapShot
Data Collection Record Prefix					
24	18	duraTime	4	Integer	Interval duration time for subsystem (in milliseconds)
28	1C	subsysName	8	Character	Subsystem name
36	24	endTime	4	Packed	Interval end time in the form 0*hhmmss*F. This is the packed decimal time in hexadecimal format (the SAS RMFSTAMPW format).
40	28	endDate	4	Packed	Interval end date in the form 0*cyyddd*F where: c=0 for 1990 to 1999 c=1 for 2000 to 2099 This is the packed decimal date in hexadecimal format (the SAS RMFSTAMPW format).
44	2C	subDurat	4	Integer	Interval duration time for subsystem (in milliseconds)—to be used in calculating overall subsystem rates
48	30	activPrt	1	Character	Partitions active during this collection interval T = Test P = Production B = Test and Production
49	31	version	1	Character	Version of this record subtype 1 = current version
50	32	(reserved)	2	Bitstring	Reserved
52	34	collId	16	Character	Collection run identifier
68	44	cpuID	8	Character	SMFID padded with blanks
76	4C	(reserved)	4	Bitstring	Reserved

Subsystem Performance Record

As shown in Figure 10-2, the Subsystem Performance record consists of:

1. The SMF record header (24 bytes)
2. The Data Collection record prefix (56 bytes)
3. Two Information Segment Descriptors—one for each Information Segment
4. The Subsystem Information Segment
5. The Space Effectiveness Information Segment

There is always one Subsystem Information Segment; there may be one Space Effectiveness Information Segment or there may be none.

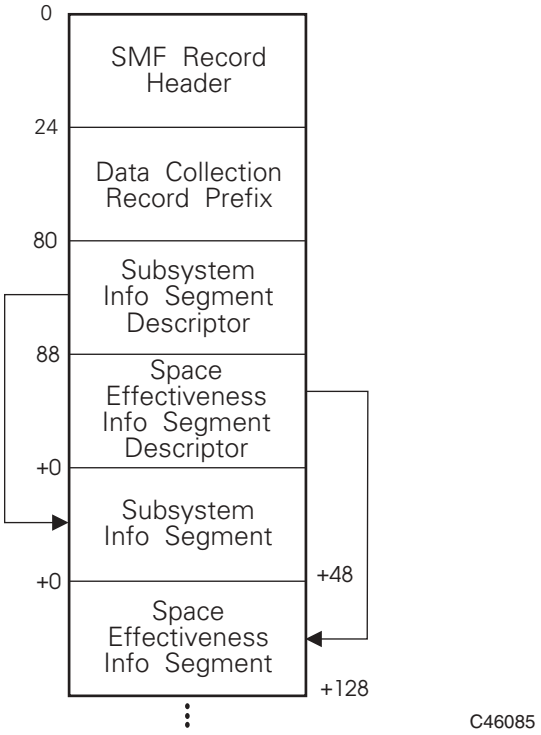


Figure 10-2. *Subsystem Performance record*

Table 10-2 shows the format of the Subsystem Performance record (subtype 1).

Table 10-2 (Page 1 of 2). *Subsystem Performance record (SMF record subtype 1)*

Dec Offset	Hex Offset	Field Name	Length	Type	Description
SMF Record Header and Data Collection Record Prefix					
0	00		80		See Table 10-1 on page 10-3
Subsystem Information Segment Descriptor					
80	50	infOffst	4	Integer	Offset to Subsystem Information Segment
84	54	infSegSz	2	Integer	Size of Subsystem Information Segment
86	56	infSegCt	2	Integer	Number of Subsystem Information Segments in this record -- always 1

Table 10-2 (Page 2 of 2). *Subsystem Performance record (SMF record subtype 1)*

Dec Offset	Hex Offset	Field Name	Length	Type	Description
			Space Effectiveness Information Segment Descriptor		
88	58	spcOffst	4	Integer	Offset to Space Effectiveness Information Segment for the subsystem
92	5C	spcSegSz	2	Integer	Size of a Space Effectiveness Information Segment
94	5E	spcSegCt	2	Integer	Number of Space Effectiveness Information Segments in this record -- can be 0 or 1
			Subsystem Information Segment		
+0	00	custCach	4	Integer	Cache size (MB)
+4	04	offCach	4	Integer	Offline cache (MB)
+8	08	pindCach	4	Integer	Pinned cache (MB)
+12	0C	nvsSize	4	Integer	NVS size (MB)
+16	10	rGrpCnt	2	Integer	Array count 0 = No 1 = Yes
+18	12	iceberg	1	Character	This is an SVA device
+19	13	(reserved)	1	Character	Reserved
+20	14	ecamMsgs	4	Integer	Count of ECAM messages processed
+24	18	ecamNspc	4	Integer	Number of ECAM channel programs bypassed because no buffer space was available
+28	1C	ecamCfBs	4	Integer	Number of ECAM channel programs bypassed because no configuration was busy
+32	20	ecamPgms	4	Integer	Number of ECAM channel programs
+36	24	(reserved)	12	Character	Reserved
			Space Effectiveness Information Segment		
+0	00	totBeCpT	8	Double	Total back-end capacity of Test partition (in bytes)
+8	08	totBeCpP	8	Double	Total back-end capacity of Production partition (in bytes)
+16	10	freBeCpT	8	Double	Free back-end capacity in Test partition available for user data (in bytes) -- includes capacity reserved for standard volumes
+24	18	freBeCpP	8	Double	Free back-end capacity in Production partition available for user data (in bytes) -- includes capacity reserved for standard volumes
+32	20	fSpcBtRT	8	Double	Total number of bytes read for free space collection in Test partition
+40	28	fSpcBtRP	8	Double	Total number of bytes read for free space collection in Production partition
+48	30	fSpcColT	8	Double	Total free space collected in Test partition
+56	38	fSpcColP	8	Double	Total free space collected in Production partition
+64	40	(reserved)	8	Character	Reserved
+72	48	stndDefd	8	Double	Standard capacity defined
+80	50	freBeSct	8	Double	Free collected back-end space
+88	58	freBeScp	8	Double	Free collected back-end space in Production partition
+96	60	(reserved)	32	Character	Reserved in Test partition

Channel Interface Statistics Record

As shown in Figure 10-3, the Channel Interface Statistics record consists of:

1. The SMF record header (24 bytes)
2. The Data Collection record prefix (56 bytes)
3. Information Segment Descriptor—one for each Information Segment
4. Channel Interface Statistics Information Segments—one per attached channel

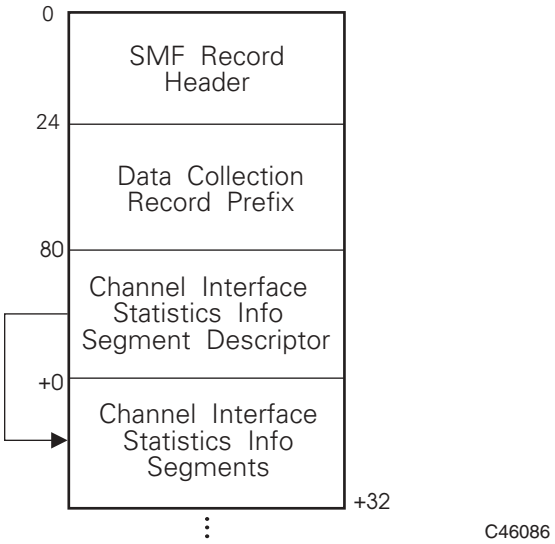


Figure 10-3. *Channel Interface Statistics record*

Table 10-3 shows the format of the Channel Interface Statistics record (subtype 2).

Table 10-3 (Page 1 of 2). *Channel Interface Statistics record (SMF record subtype 2)*

Dec Offset	Hex Offset	Field Name	Length	Type	Description
SMF Record Header and Data Collection Record Prefix					
0	00		80		See Table 10-1 on page 10-3.
Channel Interface Statistics Information Segment Descriptor					
80	50	chnOffst	4	Integer	Offset to Channel Interface Statistics Segment
84	54	chnSegSz	2	Integer	Size of a Channel Interface Statistics Segment
86	56	chnSegCt	2	Integer	Number of Channel Interface Segments in this record
Channel Interface Statistics Information Segment					
+0	00	cluster	1	Character	Channel cluster (0 or 1)
+1	01	intfId	1	Character	Channel interface identifier (A through P)
+2	02	intfName	8	Character	Channel interface name
+10	0A	intEnCur	2	Integer	Channel interface currently enabled 0 = No 1 = Yes
+12	0C	chanSped	2	Integer	Channel speed -- 3.0, 4.5, or 20.0 megabytes/second
+14	0E	(reserved)	2	Character	Reserved
+16	10	numberIo	4	Integer	Number of I/Os
+20	14	tiCoBuCh	4	Integer	Time control unit busy at channel (in milliseconds)

Table 10-3 (Page 2 of 2). *Channel Interface Statistics record (SMF record subtype 2)*

Dec Offset	Hex Offset	Field Name	Length	Type	Description
+24	18	chanType	2	Integer	Channel interface type 0 = uninstalled 1 = parallel 2 = ESCON 3 = SCSI 4 = fibre 6 = FICON
+26	1A	(reserved)	6	Integer	Reserved

Functional Device Performance Record

As shown in Figure 10-4, the Functional Device Performance record consists of:

1. The SMF record header (24 bytes)
2. The Data Collection record prefix (56 bytes)
3. Three Information Segment Descriptors—one for each Information Segment
4. Three Information Segments

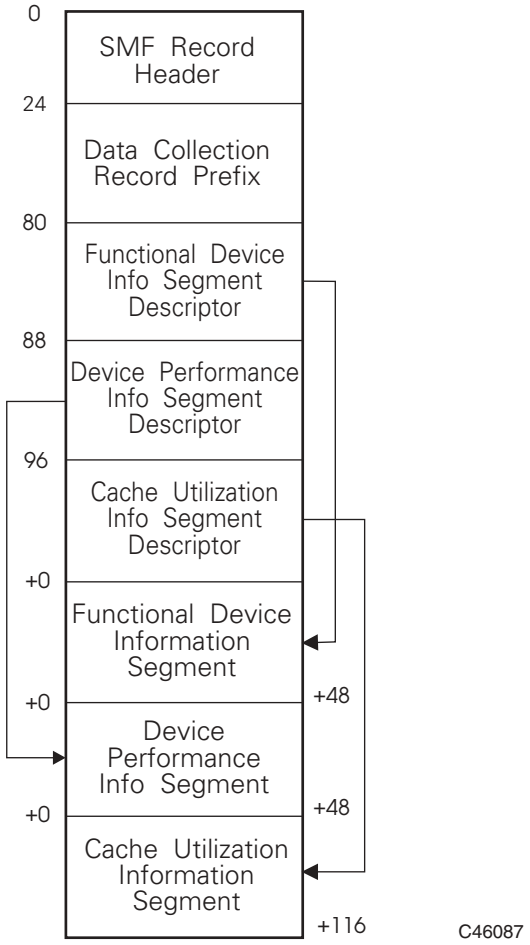


Figure 10-4. *Functional Device Performance record*

Table 10-4 shows the format of the Functional Device Performance record (subtype 3).

Table 10-4 (Page 1 of 3). *Functional Device Performance record (SMF record subtype 3)*

Dec Offset	Hex Offset	Field Name	Length	Type	Description
SMF Record Header and Data Collection Record Prefix					
0	00		80		See Table 10-1 on page 10-3.
Functional Device Information Segment Descriptor					
80	50	infOffst	4	Integer	Offset to Functional Device Information Segment
84	54	infSegSz	2	Integer	Size of Functional Device Information Segment
86	56	infSegCt	2	Integer	Number of Functional Device Information Segments in this record -- always 1.
Device Performance Information Segment Descriptor					
88	58	prfOffst	4	Integer	Offset to Device Performance Segment.
92	5C	prfSegSz	2	Integer	Size of a Device Performance Segment
94	5E	prfSegCt	2	Integer	Number of Device Performance Segments in this record -- can be 0 or 1
Cache Utilization Information Segment Descriptor					
96	60	cacOffst	4	Integer	Offset to Cache Utilization Information Segment
100	64	cacSegSz	2	Integer	Size of a Cache Utilization Information Segment
102	66	cacSegCt	2	Integer	Number of Cache Utilization Information Segments in this record -- can be 0 or 1
Functional Device Information Segment					
+0	00	vdid	2	Integer	Functional device identifier (0-4095)
+2	02	partMemb	1	Character	Partition membership T = Test P = Production
+3	03	devValid	1	Integer	Device number (devNum) is valid flag 0 = No 1 = Yes
+4	04	devNum	2	Integer	Device address as known to the host SCP
+6	06	volSer	6	Character	Volume serial number
+12	0C	vDevName	8	Character	Device name
+20	14	iceberg	1	Character	This is an SVA device 0 = No 1 = Yes
+21	15	(reserved)	3	Character	Reserved
+24	18	readReqs	4	Integer	Total read request count
+28	1C	writReqs	4	Integer	Total write request count
+32	20	devActiv	4	Integer	Number of I/O operations (number of end-of-chain events)
+36	24	vDevType	6	Character	Type of virtual device
+42	2A	volType	2	Integer	Type of volume 0 = CKD (includes PAV Base devices) 1 = CKD with SCSI format 2 = CKD Power PPRC Data Bridge 3 = CKD Power PPRC Status Bridge
+44	2C	(reserved)	4	Character	Reserved

Table 10-4 (Page 2 of 3). *Functional Device Performance record (SMF record subtype 3)*

Dec Offset	Hex Offset	Field Name	Length	Type	Description
			Device Performance Information Segment		
+0	00	feBytXfR	8	Double	Count of front-end bytes transferred -- read operations
+8	08	feBytXfW	8	Double	Count of front-end bytes transferred -- write operations
+16	10	availTim	4	Integer	Device available time (in milliseconds)
+20	14	devUtlTi	4	Integer	Device utilization time (in milliseconds)
+24	18	connTime	4	Integer	Device connect time (in milliseconds)
+28	1C	(reserved)	4	Integer	Reserved
+32	20	primCapa	4	Integer	Primary capacity in cylinders
+36	24	trackCyl	4	Integer	Tracks per cylinder
+40	28	byteTrak	4	Integer	Bytes per track
+44	2C	scsilbt	2	Integer	SCSI logical blocks per track
+46	2E	scsilbsz	2	Integer	SCSI logical block size
			Cache Utilization Information Segment		
+0	00	(reserved)	4	Integer	Reserved
+4	04	rdNorReq	4	Integer	Search or read normal request count
+8	08	rdNorHit	4	Integer	Search or read normal hits
+12	0C	wrNorReq	4	Integer	Write normal requests
+16	10	dfwNorHt	4	Integer	DASD Fast Write normal hits
+20	14	rdSeqReq	4	Integer	Search or read sequential request count
+24	18	rdSeqHit	4	Integer	Search or read sequential hits
+28	1C	wrSeqReq	4	Integer	Write sequential requests
+32	20	dfwSeqHt	4	Integer	DASD Fast Write sequential hits
+36	24	rdCFWreq	4	Integer	Search or read Cache Fast Write requests
+40	28	rdCFWhit	4	Integer	Search or read Cache Fast Write hits
+44	2C	wrCFWreq	4	Integer	Cache Fast Write requests
+48	30	wrCFWhit	4	Integer	Cache Fast Write hits
+52	34	inhCaLrq	4	Integer	Inhibit cache loading requests
+56	38	bypCaReq	4	Integer	Bypass cache requests
+60	3C	caSeqXfr	4	Integer	Sequential DASD to cache transfers (stages)
+64	40	caXfrStg	4	Integer	DASD to cache transfers (stages)
+68	44	caDasdXf	4	Integer	Cache to DASD transfers
+72	48	dfwNVSct	4	Integer	DASD Fast Write NVS constraint count
+76	4C	dfwNorWr	4	Integer	DASD Fast Write normal write requests
+80	50	dfwSeqWr	4	Integer	DASD Fast Write sequential write requests
+84	54	(reserved)	4	Integer	Reserved
+88	58	seqIntAc	4	Integer	Sequential-detected sequential access read requests
+92	5C	caEnbCur	1	Integer	Device caching status 0 = Caching inactive for device 1 = Caching active for device

Table 10-4 (Page 3 of 3). *Functional Device Performance record (SMF record subtype 3)*

Dec Offset	Hex Offset	Field Name	Length	Type	Description
+93	5D	dfwEnCur	1	Integer	Device DASD Fast Write status 0 = DASD Fast Write inactive for device 1 = DASD Fast Write active for device
+94	5E	(reserved)	2	Integer	Reserved
+96	60	lowRfCnt	4	Integer	Low reference count
+100	64	lowRfLRU	4	Integer	Low reference count due to LRU
+104	68	frDealCt	4	Integer	Frame deallocation count
+108	6C	caTrkSec	4	Integer	Cache occupancy (track milliseconds)
+112	70	(reserved)	4	Integer	Reserved

Drive Module Performance Record

Figure 10-5 shows the layout of the Drive Module Performance record, which consists of:

1. The SMF record header (24 bytes)
2. The Data Collection record prefix (56 bytes)
3. Information Segment Descriptor—one for each Information Segment
4. Drive Module Performance Information Segments

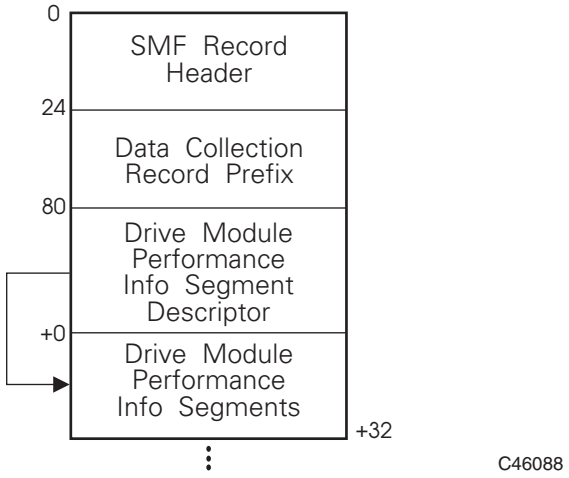


Figure 10-5. *Drive Module Performance record*

Table 10-5 on page 10-12 shows the format of the Drive Module Performance record (subtype 4).

Drive Module Performance Record

Table 10-5. *Drive Module Performance record (SMF record subtype 4)*

Dec Offset	Hex Offset	Field Name	Length	Type	Description
SMF Record Header and Data Collection Record Prefix					
0	00		80		See Table 10-1 on page 10-3.
Drive Module Performance Information Segment Descriptor					
80	50	prfOffst	4	Integer	Offset to Drive Module Performance Segment
84	54	prfSegSz	2	Integer	Size of a Drive Module Performance Segment
86	56	prfSegCt	2	Integer	Number of Drive Module Performance Segments in this record
Drive Module Performance Information Segment					
+0	00	unit	1	Integer	Unit
+1	01	tray	1	Integer	Tray
+2	02	slot	1	Integer	Slot
+3	03	partMemb	1	Character	Partition membership (T = Test, P = Production)
+4	04	drvMdTim	4	Integer	Drive module interval duration (in milliseconds)
+8	08	busyTime	4	Integer	Drive module busy time (in milliseconds)
+12	0C	(reserved)	4	Integer	Reserved
+16	10	readByts	8	Excess 64[1]	Bytes transferred -- read operations
+24	18	wrteByts	8	Excess 64	Bytes transferred -- write operations

[1] Excess 64 notation is the long floating-point notation in which the characteristic of a floating-point number is represented by its binary equivalent plus 64. The first two hexadecimal digits represent the sign of the fraction (the first bit) and the exponent. The other 6 hexadecimal digits represent the mantissa of the fraction (which is always positive).

Space Utilization Record

The Space Utilization record consists of:

1. The SMF record header (24 bytes)
2. The basic segment
3. The report summary heading
4. The first section: space utilization report summary
5. The second section: selected devices summary
6. The third section: space utilization summary

In general, the fields of the space utilization SMF event record correspond to the fields of the space utilization report (see “Sample Space Utilization Report” on page 9-11). However, the SMF record also includes fields that are used for internal calculations and are not shown in the report.

For each field in the event record shown in Table 10-6, the description column entry begins with one of:

- (***) to indicate a calculation field; these do not appear in the report
- The heading (in all caps) that appears in the report
- An indication that there is no corresponding heading in the report

Table 10-6 (Page 1 of 5). *Space Utilization event record (SMF record subtype 7)*

Dec Offset	Hex Offset	Field Name	Length	Type	Description
			SMF Record Header		
0	00	len	2	Binary	Record length. This field and the next make up the record descriptor word.
2	02	seg	2	Binary	Segment descriptor
4	04	smfflg	1	Binary	X'80' New SMF record format X'40' Subtype utilized X'20' (reserved) X'10' MVS/SP Version 4 X'08' MVS/ESA X'04' MVS/XA X'02' OS/VS2 X'01' (reserved)
5	05	smfrty	1	Binary	SMF record number
6	06	smftime	4	Binary	Time record written, in hundredths of seconds since midnight
10	0A	smfdate	4	Packed	Date record written, in the form 0*cyyddd*F where: c=0 for 1990 to 1999 c=1 for 2000 to 2099
14	0E	smfsid	4	Character	System ID
18	12	smfssi	4	Character	Subsystem identification for SMF
22	16	rectype	2	Binary	SMF record subtype X'0001' Subsystem Performance X'0002' Channel Interface Statistics X'0003' Functional Device Performance X'0004' Drive Module Performance X'0005' DDSR X'0006' (reserved) X'0007' Space Utilization X'0008' SnapShot

Table 10-6 (Page 2 of 5). *Space Utilization event record (SMF record subtype 7)*

Dec Offset	Hex Offset	Field Name	Length	Type	Description
			Basic Segment		
0	00	ssucode	2	Integer	(***) LIC level
2	02	totflag	2	Binary	(***) Totals record flag: 1 = This is a totals record 2 = Not a totals record If this is a totals record, all fields except the totals fields are blank.
4	04	noTrMapd	4	Integer	(***) Number of tracks mapped
8	08	primCapa	2	Integer	Primary capacity in cylinders
10	0A	trackCyl	2	Integer	(***) Tracks per cylinder
12	0C	byteTrak	4	Integer	(***) Bytes per track
16	10	capntstr	8	Double	Functional capacity not stored
24	18	spAlloc	8	Double	(***) Space allocated
32	20	fnctcap	8	Double	(***) Functional capacity (primCapa*trackCyl*byteTrak)
40	28	capstord	8	Double	(***) Capacity stored (noTrMapd*byteTrak)
48	30	beBytExt	8	Double	(***) Number of back-end bytes
56	38	totBeCpT	8	Double	(***) Total back-end capacity in Test partition
64	40	totBeCpP	8	Double	(***) Total back-end capacity in Production partition
72	48	freBeCpT	8	Double	(***) Free back-end capacity in Test partition
80	50	freBeCpP	8	Double	(***) Free back-end capacity in Production partition
88	58	freBeScT	8	Double	(***) Free back-end space collected in Test partition
96	60	freBeScP	8	Double	(***) Free back-end space collected in Production partition
			Space Utilization Summary Heading		
+104	68	subSName	9	Character	SUBSYSTEM: Subsystem name
+113	71	res1	1	Character	Reserved
+114	72	fDevCnt	2	Integer	NUMBER OF FUNCTIONAL DEVICES: Number of functional devices
			Space Utilization Report Summary		
+116	74	vDevName	9	Character	FDID: Name of functional device
+125	7D	devNum	5	Character	DEV ADDR: Device address
+130	82	volSer	9	Character	VOLSER: Volume serial number
+139	8B	res2	1	Character	Reserved
+140	8C	offflag	2	Character	(No heading in report.) Offline flag: if an asterisk appears between the VOLSER and T/P columns in the report, the volume is offline
+142	8E	errflg	2	Character	(No heading in report.) Device error flag: if there is a device error, ALLOC fields in the report are changed to “V-ERR.”
+144	90	partMemb	2	Character	T/P: Partition membership (Test or Production)
+146	92	vDevType	7	Character	DEVICE TYPE: Type of virtual device
+153	99	res3	7	Character	(Reserved)
+160	A0	fcapcity	8	Double	FUNCT CAP: Functional capacity in megabytes

Table 10-6 (Page 3 of 5). *Space Utilization event record (SMF record subtype 7)*

Dec Offset	Hex Offset	Field Name	Length	Type	Description
					FUNCTIONAL CAPACITY
+168	A8	falloc	8	Double	ALLOC: Functional capacity allocated (in megabytes)
+176	B0	fstored	8	Double	STORED: Functional capacity stored (in megabytes)
+184	B8	fnotstor	8	Double	NOT STORED: Functional capacity not stored (in megabytes)
					SELECTED DEVICES
+192	C0	palloc	8	Double	ALLOC: Percentage of functional capacity allocated
+200	C8	pstored	8	Double	STORED: Percentage of functional capacity stored
+208	D0	pnotstor	8	Double	NOT STORED: Percentage of functional capacity not stored
+216	D8	PhCapUse	8	Double	PHYS CAP USED: Physical capacity used (megabytes)
+224	E0	cmprat	8	Double	COMPRESS RATIO: Compression ratio
+232	E8	backstor	1	Character	(No heading in report.) If PhCapUse = 0.0 but the value is greater than zero before rounding, "*" is appended to the PHYS CAP USE field.
+233	E9	res4	1	Character	(Reserved)
					Selected Devices Summary
					SELECTED DEVICES
+234	EA	tDevCntP	2	Integer	PRODUCTION PARTITION: Total number of functional devices -- Production partition
+236	EC	tDevCntT	2	Integer	TEST PARTITION: Total number of functional devices -- Test partition
+238	EE	tDevCntB	2	Integer	TOTALS:) Total number of functional devices -- both partitions
					TOTAL FUNCTIONAL CAPACITY
+240	F0	tFunCapP	8	Double	PRODUCTION PARTITION: Total functional capacity (in megabytes) -- Production partition
+248	F8	tFunCapT	8	Double	TEST PARTITION: Total functional capacity (in megabytes) -- Test partition
+256	100	tFunCapB	8	Double	TOTALS: Total functional capacity (in megabytes) -- both partitions
					TOTAL FUNCTIONAL CAPACITY STORED
+264	108	tFuCpStP	8	Double	PRODUCTION PARTITION: Total functional capacity stored (in megabytes) -- Production partition
+272	110	tFuCpStT	8	Double	TEST PARTITION: Total functional capacity stored (in megabytes) -- Test partition
+280	118	tFuCpStB	8	Double	TOTALS: Total functional capacity stored (in megabytes) -- total of both partitions
					TOTAL FUNCTIONAL CAPACITY NOT STORED
+288	120	tFuCpNoP	8	Double	PRODUCTION PARTITION: Total functional capacity not stored (in megabytes) -- Production partition
+296	128	tFuCpNoT	8	Double	TEST PARTITION: Total functional capacity not stored (in megabytes) -- Test partition
+304	130	tFuCpNoB	8	Double	TOTALS: Total functional capacity not stored (in megabytes) -- total of both partitions

Table 10-6 (Page 4 of 5). *Space Utilization event record (SMF record subtype 7)*

Dec Offset	Hex Offset	Field Name	Length	Type	Description
					% FUNCTIONAL CAPACITY STORED
+312	138	pFuCpStP	8	Double	PRODUCTION PARTITION: Percentage functional capacity stored -- Production partition
+320	140	pFuCpStT	8	Double	TEST PARTITION: Percentage functional capacity stored -- Test partition
+328	148	pFuCpStB	8	Double	TOTALS: Percentage functional capacity stored -- both partitions
					% FUNCTIONAL CAPACITY NOT STORED
+336	150	pFuCoNoP	8	Double	PRODUCTION PARTITION: Percentage functional capacity not stored -- Production partition
+344	158	pFuCoNoT	8	Double	TEST PARTITION: Percentage functional capacity not stored -- Test partition
+352	160	pFuCpNoB	8	Double	TOTALS: Percentage functional capacity not stored -- both partitions
					PHYS CAP USED
+360	168	tPhCpUsP	8	Double	PRODUCTION PARTITION: Total physical capacity of disk array used -- Production partition
+368	170	tPhCpUsT	8	Double	TEST PARTITION: Total physical capacity of disk array used -- Test partition
+376	178	tPhCpUsB	8	Double	TOTALS: Total physical capacity of disk array used -- both partitions
					COMPRESS RATIO
+384	180	tCmpRatP	8	Double	PRODUCTION PARTITION: Compression ratio of disk array -- Production partition
+392	188	tCmpRatT	8	Double	TEST PARTITION: Compression ratio of disk array -- Test partition
+400	190	tCmpRatB	8	Double	TOTALS: Compression ratio of disk array -- both partitions
+408	198	tBakStoP	1	Character	(No heading in report.). Flag (in report, asterisk right of PHYS CAP USED column) to indicate that physical capacity of disk array used is greater than 1.0 -- Production partition
+409	199	tBakStoT	1	Character	(No heading in report.) Flag (in report, asterisk right of PHYS CAP USED column) to indicate that physical capacity of disk array used is greater than 1.0 -- Test partition
+410	19A	tBakStoB	1	Character	(No heading in report.) Flag (in report, asterisk right of PHYS CAP USED column) to indicate that physical capacity of disk array used is greater than 1.0 -- both partitions
+411	19B	res5	5	Character	(Reserved)
					Space Utilization Summary
+416	1A0	totlpcap	8	Double	DISK ARRAY CAPACITY: Capacity of disk array (in megabytes)
					NET CAPACITY LOAD %
+424	1A8	ncapldpt	8	Double	TEST: Percentage net capacity load -- Test partition
+432	1B0	ncapldpp	8	Double	PROD: Percentage net capacity load -- Production partition

Table 10-6 (Page 5 of 5). *Space Utilization event record (SMF record subtype 7)*

Dec Offset	Hex Offset	Field Name	Length	Type	Description
+440	1B8	ncapldp	8	Double	OVERALL: Percentage net capacity load -- overall
					COLL FREE SPACE %
+448	1C0	fspccolt	8	Double	TEST: Percentage collected free space -- Test partition
+456	1C8	fspccolp	8	Double	PROD: Percentage collected free space -- Production partition
+464	1D0	fspccol	8	Double	OVERALL: Percentage collected free space -- overall
					UNCOLL FREE SPACE %
+472	1D8	fspcunct	8	Double	TEST: Percentage uncollected free space -- Test partition
+480	1E0	fspcuncp	8	Double	PROD: Percentage uncollected free space -- Production partition
+488	1E8	fspcunc	8	Double	OVERALL: Percentage uncollected free space -- overall
+496	1F0	datetime	25	Character	(***) Date and time record was produced
+521	209	res6	3	Character	(Reserved)
+524	20C	noTrMapu	4	Integer	(***) Number of unique tracks mapped
+528	210	beBytUnq	8	Double	(***) Number of unique back end bytes
+536	218	beBytShr	8	Double	(***) Number of shared back end bytes
+544	220	PhCapUsS	8	Double	Shared physical cap used (in megabytes)
+552	228	PhCapUsU	8	Double	Unique physical cap used (in megabytes)
+560	230	tPhCpSrP	8	Double	PRODUCTION PARTITION: Total shared physical capacity used -- Production partition
+568	238	tPhCpSrT	8	Double	TEST PARTITION: Total shared physical capacity used -- Test partition
+576	240	tPhCpSrB	8	Double	TOTALS: Total shared physical capacity used -- both partitions
+584	248	tPhCpUnP	8	Double	PRODUCTION PARTITION: Total unique physical capacity used -- Production partition
+592	250	tPhCpUnT	8	Double	TEST PARTITION: Total unique physical capacity used -- Test partition
+600	258	tPhCpUnB	8	Double	TOTALS: Total unique physical capacity used -- both partitions
+608	260	scsilbt	2	Integer	SCSI logical blocks per track
+610	262	scsilbsz	2	Integer	SCSI logical block size
+612	264	volType	2	Integer	Type of volume 0 = CKD (includes PAV Base devices) 1 = CKD with SCSI format 2 = CKD Power PPRC Data Bridge 3 = CKD Power PPRC Status Bridge
+614	266	res7	26	Character	(Reserved)

Space Utilization Record

Appendix A. Sample OS/390 Job Control Language

Summary

This appendix contains sample job control language (JCL) for Reporter.

In the sample JCL, text within **< >** symbols is explanatory and must be replaced by the statements or commands appropriate for your installation.

Starting the Data Collection Task

Use the following JCL to start the Data Collection Task for a subsystem.

```
//ssname  PROC MEM=SIBSYSxx
//IEFPROC EXEC PGM=SIBMAIN,
//    PARM='&MEMBER'
//    TIME=1440,
//    REGION=2048K
//*
//* SVAA LOAD LIBRARY
//STEPLIB DD DSN=hlq.SIBLOAD,DISP=SHR
//*
//* SVAA PARAMETER LIBRARY
//STKPARMS DD DSN=hlq.sibparm,DISP=SHR
//         DD DSN=hlq.rmfparms,DISP=SHR
//SIBCTRAN DD DSN=hlq.SACLINK,DISP=SHR
//CTRANS   DD DSN=hlq.SACLINK,DISP=SHR
//* STANDARD SYSOUT
//SYSPRINT DD SYSOUT=A
//SYSTERM  DD SYSOUT=A
/*
//* REPORTER DATA COLLECTION FILES
//outfile1 DD DSN=filename1,DISP=SHR
//outfile2 DD DSN=filename2,DISP=SHR
//*
//* OPTIONAL DATA SET FOR DUMPS FROM SVAA STARTED TASK
//SYSMDUMP DD DSN=filename3,DISP=SHR
//*
```

The name on the **PROC** statement must match one of the SVAA OS/390 subsystem names defined in the OS/390 Subsystem Name Table (see the *SVAA for OS/390 Installation, Customization, and Maintenance* manual).

The **STKPARMS DD** statement points to a library containing a member (SIBSYS*nn*) with SVAA startup parameters for the SVAA address space. The MEMBER symbolic parameter specifies the suffix of the member containing the SVAA startup parameters. One of the statements in that member must be SRP=*xx*. This statement tells SVAA to read the SIBSRP*xx* member to obtain the startup parameters (such as the **SET ECAMDEVICE** statement).

STKPARMS can also be used for the SVAA RMF parameter values, using member ERBRMF*nn*. (This can be the same data set as a concatenated data set.)

The **STKPARMS DD** statement is optional. If the **STKPARMS DD** statement is missing, SVAA looks in SYS1.PARMLIB for the **SIBSYS***xx* member and the **ERBRMF***nn* member.

The Reporter startup parameters can also include Reporter subcommands, such as INIT MAIN and COLL PT.

The STKPARMS library is accessed whenever the console operator issues either a **START COLLTN** command or a **SET COLLTN** command with the **MEMBER** parameter to modify an existing data-collection run.

If the *hlq*.**SACLINK** data set is not in the linklist, use either the **SIBCTRAN** or **CTRANS DD** statement. For an example of the use of **SIBCTRAN**, see the SIBCTUM member of the SIBSAMP library.

The *outfile1* and *outfile2* **DD** statements define optional logging files that are specified on the **OUTF** or **ALTF** parameters of the **INIT MAIN** or **COLL ORD** subcommands. These statements are not needed if output is to SMF.

Use the **SYSMDUMP DD** statement if you want SVAA dumps to be readable by IPCS. Allocate the data set with the same attributes as one of your SYS1.DUMP*nn* data sets.

Summarizing the History File

Use the following JCL to summarize the history file:

```
// <jobcard>
//*      GENERATE SAS PROGRAM
//STEP1  EXEC  PGM=SIBBATCH
//STEPLIB  DD  DSN=hlq.SIBLOAD,DISP=SHR
//STKPARMS DD  DSN=hlq.sibparm,DISP=SHR
//SIBCTRAN DD  DSN=hlq.SACLINK,DISP=SHR
//CTRANS   DD  DSN=hlq.SACLINK,DISP=SHR
//SYSTERM  DD  SYSOUT=*
//SYSPRINT DD  SYSOUT=*
//SASPGM   DD  DSN=&&SASPGM,DISP=(MOD,PASS),UNIT=SYSDA,
//             SPACE=(TRK,(5,10)),
//             DCB=BLKSIZE=3120
//SYSIN    DD  *
 SUMM HIST
//*      EXECUTE SAS HISTORY FILE SUMMARIZATION PROGRAM
//* Installation's SAS Proc:
//STEP2  EXEC  SAS,OPTIONS='SASLIB=SIBLOAD,MEMSIZE=16M',COND=(0,NE)
//SIBLOAD  DD  DSN=hlq.SIBLOAD,DISP=SHR
//SYSIN    DD  DSN=&&SASPGM,DISP=(OLD,DELETE)
//DETAILIN DD  DSN=<detail input files>,DISP=SHR
//SIBCTRAN DD  DSN=hlq.SACLINK,DISP=SHR
//CTRANS   DD  DSN=hlq.SACLINK,DISP=SHR
//SIBSAS   DD  DSN=hlq.SIBSAS,DISP=SHR
//STKPERFI DD  DSN=STK.HISTORY(0),DISP=SHR
//STKPERFO DD  DSN=STK.HISTORY(+1),DISP=(NEW,CATLG,DELETE)
/*
//
```

Notes:

1. On the SASPGM DD statement, specify the appropriate UNIT for the SAS program.
2. If the *hlq*.**SACLINK** data set is not in the linklist, use either the **SIBCTRAN** or **CTRANS DD** statement. For an example of the use of **SIBCTRAN**, see the SIBCTUM member of the SIBSAMP library.

3. The default **DD** name for the detail input files is **DETAILIN**. You can modify this **DD** name by using the **DETLF** parameter of the **SUMM HIST** command.
4. The default **DD** name for the **SAS** library containing the “old” history file is **STKPERFI**, and the default **DDNAME** for the **SAS** library containing the updated history file is **STKPERFO** (note that the last character is the letter 'O'). These names can be modified by the **HISTIN** and **HISTOUT** parameters, respectively, of the **SUMM HIST** subcommand. You can provide a single **DD** statement if both the “old” and updated versions of the history file exist in the same **SAS** library. However, these files may reside on tape; in that case you must provide separate **SAS** libraries and separate **DD** statements.
5. Use the optional **PARMS='=PARM=membername'** parameter on the **SIBADMIN** command to specify a different SIBSTK*nn* member name.
6. If you are appending to an existing history file, a return code of 4 is normal.

Requesting Performance Reports

Use the following JCL to request performance reports:

```
// <jobcard>
//*      GENERATE SAS PROGRAM
//STEP1  EXEC  PGM=SIBBATCH
//STEPLIB  DD  DSN=hlq.SIBLOAD,DISP=SHR
//STKPARMS DD  DSN=hlq.PARMLIB,DISP=SHR
//SYSTERM  DD  SYSOUT=*
//SYSPRINT DD  SYSOUT=*
//SASPGM   DD  DSN=&&SASPGM,DISP=(MOD,PASS),UNIT=SYSDA,
//             SPACE=(TRK,(5,10)),
//             DCB=BLKSIZE=3120
//EXCPTRPT DD  DSN=<optional threshold file>,DISP=SHR
//SIBCTRAN DD  DSN=hlq.SACLINK,DISP=SHR
//CTRANS   DD  DSN=hlq.SACLINK,DISP=SHR
//SYSIN    DD  *
 REPORT PERF
//*      EXECUTE SAS REPORT PROGRAM
//*  SAS PROC
//STEP2  EXEC  SAS,OPTIONS='SASLIB=SIBLOAD,MEMSIZE=16M',COND=(4,LT)
//SIBLOAD  DD  DSN=hlq.SIBLOAD,DISP=SHR
//SYSIN    DD  DSN=&&SASPGM,DISP=(OLD,DELETE)
//SYSTERM  DD  DSN=SYSOUT=*
//SIBCTRAN DD  DSN=hlq.SACLINK,DISP=SHR
//CTRANS   DD  DSN=hlq.SACLINK,DISP=SHR
//*  SVAA SAS MACRO LIBRARY
//SIBSAS   DD  DSN=hlq.SIBSAS,DISP=SHR
//*
//RPTIN    DD  <report input files>,DISP=SHR
/*
//
```

Notes:

1. On the SASPGM DD statement, specify the appropriate UNIT for the SAS program.
2. The **MEMSIZE** subparameter is optional and overrides the system default for the SAS MEMSIZE option.
3. If the *hlq*.**SACLINK** data set is not in the linklist, use either the **SIBCTRAN** or **CTRANS DD** statement. For an example of the use of **SIBCTRAN**, see the SIBCTUM member of the SIBSAMP library.
4. **DD** statements are required for any **SAS** libraries specified by the **SASDFL** or **SASSFL** parameters of the **REPORT PERF** command.
5. The default **DD** name for the report input files is **RPTIN**. You can modify this **DD** name by using the **RPTF** parameter of the **REPORT PERF** subcommand. If the input is the history file, this **DD** statement specifies the **SAS** library containing the file.
6. Use the optional **PARMS='=PARM=membername'** parameter on the **SIBADMIN** command to specify a different SIBSTK*nn* member name.

Requesting Space Utilization Reports

Use the following JCL to request space utilization reports:

```
// <jobcard>
//*      GENERATE SAS PROGRAM
//STEP1  EXEC  PGM=SIBBATCH
//STEPLIB  DD  DSN=hlq..SIBLOAD,DISP=SHR
//STKPARMS DD  DSN=hlq.PARMLIB,DISP=SHR
//SYSTERM  DD  SYSOUT=*
//SYSPRINT DD  SYSOUT=*
//SASPGM   DD  DSN=&&SASPGM,DISP=(MOD,PASS),UNIT=SYSDA,
//             SPACE=(TRK,(5,10)),
//             DCB=BLKSIZE=3120
//SIBCTRAN DD  DSN=hlq.SACLINK,DISP=SHR
//CTRANS   DD  DSN=hlq.SACLINK,DISP=SHR
//SYSIN    DD  *
 REPORT SPACEU
//*      EXECUTE SAS REPORT PROGRAM
//*  SAS PROC
//STEP2  EXEC  SAS,OPTIONS='SASLIB=SIBLOAD MEMSIZE=16M',COND=(4,LT)
//SIBLOAD  DD  DSN=hlq..SIBLOAD,DISP=SHR
//SIBCTRAN DD  DSN=hlq.SACLINK,DISP=SHR
//CTRANS   DD  DSN=hlq.SACLINK,DISP=SHR
//SYSIN    DD  DSN=&&SASPGM,DISP=(OLD,DELETE)
//*  SVAA SAS MACRO LIBRARY
//SIBSAS   DD  DSN=hlq.SIBSAS,DISP=SHR
/*
//
```

Notes:

1. On the SASPGM DD statement, specify the appropriate UNIT for the SAS program.
2. The **MEMSIZE** subparameter is optional and overrides the system default for the SAS MEMSIZE option.
3. If the *hlq*.**SACLINK** data set is not in the linklist, use either the **SIBCTRAN** or **CTRANS DD** statement. For an example of the use of **SIBCTRAN**, see the SIBCTUM member of the SIBSAMP library.
4. Use the optional **PARMS='=PARM=membername'** parameter on the **SIBADMIN** command to specify a different SIBSTK*nn* member name.

Copying Full Logging File Procedure (SIBOFFLD)

You should install the following procedure in a standard proclib and tailor the variables (for example, names and space allocation) for your environment.

```
//SIBOFFLD PROC FILENAM=,       NAME OF FILE TO OFFLOAD
//         PRIME=20,            OFFLOADED FILE PRIMARY CYLS
//         SEC=2                OFFLOADED FILE SECONDARY CYLS
//*
//* COPY THE REPORTER COLLECTION FILE
//*
//STEP1  EXEC  PGM=IEBGENER
//SYSPRINT DD  SYSOUT=A
//SYSUT1   DD  DSN=&FILENAM,DISP=SHR
//SYSUT2   DD  DSN=hlq.XSARPT.DATA(+1),DISP=(,CATLG,DELETE),
//         UNIT=SYSDA,SPACE=(CYL,(&PRIME,&SEC),RLSE)
//SYSIN    DD  DUMMY
//*
//* MAKE THE REPORTER OUTPUT FILE EMPTY
//*
//STEP2  EXEC  PGM=IEBGENER
//SYSPRINT DD  SYSOUT=A
//SYSUT1   DD  DUMMY,DCB=&FILENAM
//SYSUT2   DD  DSN=&FILENAM,DISP=SHR
//SYSIN    DD  DUMMY
//         PEND
```

Notes:

1. If the SYSUT2 statement names a GDG, you must create the GDG. If the sample DSCB does not contain DCB parameters, you must add them to the JCL.
2. To offload to tape rather than to disk, remove the disk allocation parameters and replace the disk information in the SYSUT2 statement with unit, label, and retention period information for copying to tape.

Executing the Program to Produce Comma-Delimited Output

The function that produces a data extraction program generates comma-delimited output with the following JCL:

```
// <jobcard>
//* EXAMPLE OF JCL GENERATED WHEN CREATING COMMA-DELIMITED OUTPUT
//*
//*      INSTALLATIONS SAS PROC
//STEP1  EXEC  SAS,COND=(0,NE),OPTIONS='SASLIB=SIBLOAD S=0'
//WORK     DD  SPACE=(CYL,(30,5))
//SIBLOAD  DD  DSN=hlq.SIBLOAD,DISP=SHR
//SYSIN    DD  DDNAME=SIBDXPGM
//SIBSAS   DD  DSN=hlq.SIBSAS,DISP=SHR
//SIBIN1   DD  DSN=<input history file>,DISP=SHR
//CMADTL   DD  DSN=<comma-delimited device file>,DISP=OLD
//CMATOT   DD  DSN=<comma-delimited subsys file>,DISP=OLD
//SIBDXPGM DD  *
   .
   .
  generated SAS program
   .
   .
/*
```

Executing the SAS Data Extraction Program

The function that produces a data extraction program generates SAS output with the following JCL:

```
// <jobcard>
//*  EXAMPLE OF JCL GENERATED WHEN CREATING SAS OUTPUT
//*
//*      INSTALLATIONS SAS PROC
//STEP1  EXEC  SAS,COND=(0,NE),OPTIONS='SASLIB=SIBLOAD S=0'
//WORK     DD  SPACE=(CYL,(30,5))
//SIBLOAD  DD  DSN=hlq.SIBLOAD,DISP=SHR
//SYSIN    DD  DDNAME=SIBDXPGM
//SIBSAS   DD  DSN=hlq.SIBSAS,DISP=SHR
//SIBIN1   DD  DSN=<input history file>,DISP=SHR
//SASDTL   DD  DSN=<SAS extracted device file>,DISP=OLD
//SASTOT   DD  DSN=<SAS extracted subsys file>,DISP=OLD
//XPORTDTL DD  DSN=<SAS export device file>,DISP=OLD
//XPORTTOT DD  DSN=<SAS export subsys file>,DISP=OLD
//SIBDXPGM DD  *
   .
   .
  generated SAS program
   .
   .
/*
```

Appendix B. Exception Thresholds

The following tables describe all the exception thresholds monitored by Reporter.

The first column of the tables lists subsystem variables for which you can specify reporting thresholds. The second column describes the variables. The next column shows a mnemonic abbreviation for the variable; you use these abbreviations in specifying thresholds using Reporter subcommands. The last two columns show the default values for the lower and upper thresholds of the variable. You can specify these defaults by using an asterisk for the threshold value in the panel field or subcommand parameter. Threshold values for which there is no default are marked **N/A**.

Specify the threshold names and values in one of two ways:

- As values of the THRSH parameter of the REPORT PERF subcommand or as values on the SPECIFY EXCEPTION THRESHOLDS panel.
- As entries in an exception thresholds file.

You can also specify that there be no exception reporting for a subsystem variable by specifying **NO** for that variable.

For more information, see “Specifying Exception Thresholds” on page 6-10.

Device-Performance Exception Thresholds

The following table lists the fields of the SPECIFY EXCEPTION THRESHOLDS panel that apply to Device Performance reports. You specify these thresholds using primarily the left side of the panel. The fields are ordered as you see them on the panel. (The panel appears on page 7-43.)

Table B-1 (Page 1 of 2). *Exception thresholds for device-performance reporting*

Panel field	***Description of Exception***	***Abbreviation for exception***	***LT***	***GT***
I/O rate (device)	Average number of I/O operations per second for a functional device.	IORTEDEV	N/A	40
Device Available %	Percentage of reporting interval in which the device was available (operational).	DEVAVLPCT	100	0
Drive Module Coeff Var	Coefficient of variation (expressed as a percent) of percentage drive module busy.	DRVMCOV	N/A	20
Access Density	Number of I/O operations per second per gigabyte of functional capacity defined.	ACCDEN	N/A	10
I/O Service Time	Average service time per I/O operation in milliseconds.	IOSVCTM	N/A	12
Disconnect Time	Average time device was disconnected from the channel (in milliseconds) while processing an I/O operation.	DISCTM	N/A	10
Device Utilization %	Percentage of reporting interval in which the device was utilized (busy).	UTILPCT	N/A	30
Disconnect Time %	Percentage of reporting interval in which the device was disconnected from the channel while processing an I/O operation.	DISCPCT	N/A	24

Table B-1 (Page 2 of 2). *Exception thresholds for device-performance reporting*

Panel field	*Description of Exception*	*Abbreviation for exception*	*LT*	*GT*
Channel I/O Rate	Average number of I/O operations (end-of-chain events) per second for the channel interface.	CHNLRTE	N/A	350
% Active on Channel	Percentage of reporting interval the subsystem was active (transferring data) on this channel interface.	ACTCHNPCT	N/A	50
Drive Module Util %	Average percentage drive module busy for reporting interval averaged over all drive modules.	DRVMUTILPCT	N/A	15
Net Capacity Load %	Percentage of back-end capacity that is used (not free).	NCAPLODPCT	N/A	75

Cache-Effectiveness Exception Thresholds

The following table lists the fields of the SPECIFY EXCEPTION THRESHOLDS panel that apply to Cache Effectiveness reports. You specify these thresholds using primarily the right side of the panel. The fields are ordered as you see them on the panel. (The panel appears on page 7-43.)

Table B-2 (Page 1 of 2). *Exception thresholds for cache-effectiveness reporting*

Panel field	*Description of Exception*	*Abbreviation for exception*	*LT*	*GT*
I/O rate (device)	Average number of I/O operations per second for a functional device.	IORTEDEV	N/A	40
I/O Rate (subsystem)	Average number of I/O operations per second for a subsystem.	IORTESUB	N/A	1200
I/O Hit %	Percentage of I/O operations for which the addressed track was present in cache storage.	IOHITPCT	70	N/A
Read Hit %	Percentage of read operations for which the addressed track was present in cache storage.	RDHITPCT	65	N/A
Write Hit %	Percentage of write operations for which the addressed track was present in cache storage.	WRHITPCT	90	N/A
DASD Fast Write Constraint (device)	Number of DASD Fast Write operations that were delayed (data written directly to DASD) due to insufficient nonvolatile storage, summarized by functional device.	DFWCNSTDEV	N/A	0
DASD Fast Write Constraint (subsystem)	Number of DASD Fast Write operations that were delayed (data written directly to DASD) due to insufficient nonvolatile storage, summarized by subsystem.	DFWCNSTSUB	N/A	1
Stage Rate (device)	Average number of staging operations (transfer of data from DASD storage to cache) per second, summarized by functional device	STGRTEDEV	N/A	12

Table B-2 (Page 2 of 2). *Exception thresholds for cache-effectiveness reporting*

Panel field	***Description of Exception***	***Abbreviation for exception***	***LT***	***GT***
Stage Rate (subsystem)	Average number of staging operations (transfer of data from DASD storage to cache) per second, summarized by subsystem	STGRTESUB	N/A	360
Tracks Occupied	Average number of tracks occupying cache per second per megabyte of cache, for the subsystem only.	TRKOCC	45	N/A
Hits per Stage (xx.x or xxxx)	Ratio of number of channel segments for which the addressed track was in cache storage to number of times data was transferred from DASD storage to cache. xx.x is a reminder that you can enter thresholds with at most one decimal place.	HITSTGE	2.3	N/A

Thresholds

Appendix C. Variables for Bar Chart Reports

The following table lists all subsystem variables for which you can produce bar charts.

The first column lists the subsystem variables that are available for reporting as bar charts in the order that the fields appear on the SELECT BAR CHART FIELDS panel (see Figure 7-44 on page 7-47). The second column shows the abbreviation to use in the BARCHART parameter of the REPORT PERFORMANCE subcommand to obtain a bar chart for the selected value.

Table C-1 (Page 1 of 2). *Bar chart fields*

Panel field	***Description of subsystem value***	***Abbreviation***
Device Available %	Percentage of reporting interval in which the device was available (operational).	DEVAVLPCT
Drive Module Utilization %	Average percent drive module busy for reporting interval averaged over all drive modules.	DRVMUTILPCT
I/O rate	Average number of I/O operations per second for the device, channel interface, or subsystem.	IORTE
Drive Module Coeff of Var	Coefficient of variation of percent drive module busy. This measures the evenness of real device utilization in the subsystem.	DRVMCOV
Access Density	I/O operations per second per gigabyte of functional capacity defined.	ACCDEN
I/O Hit %	Percentage of total I/O operations for which the addressed track was present in cache storage.	IOHITPCT
Kilobytes per Second	Kilobytes of data transferred between the host computer and the subsystem per second.	KBSEC
I/O Service Time	Average service time per I/O operation in milliseconds. Does not include host queueing time.	IOSVCTM
Read Hit %	Percentage of read operations for which the addressed track was present in cache storage.	RDHITPCT
Disconnect Time	Average time device was disconnected from the channel (in milliseconds) while processing an I/O operation.	DISCTM
DASD Fast Write Constraint	Percentage of DASD Fast Write operations which were bypassed (data written directly to DASD) due to of insufficient nonvolatile storage.	DFWCONSTR
Device Utilization %	Percentage of reporting interval in which the device was utilized (busy). Busy time = connect time + disconnect time.	UTILPCT
Stage Rate	Average number of staging operations (transfer of data from DASD storage to cache) per second.	STGRTE
Disconnect Time %	Percentage of reporting interval in which the device was disconnected from the channel while processing an I/O operation.	DISCPCT
Hits per Stage	Ratio of number of channel segments for which the addressed track was in cache storage to number of times data was transferred from DASD storage to cache.	HITSTGE
Free Space Collection Load	Percentage of back-end physical space collected for free space consolidation that did not yield available free space.	SPCCOLLD
Tracks Occupied	Average number of tracks occupying cache per second per megabyte of cache for the subsystem only.	TRKOCC
Net Capacity Load %	Percentage of back-end capacity that is used (not free).	NCAPLODPCT

Table C-1 (Page 2 of 2). *Bar chart fields*

Panel field	***Description of subsystem value***	***Abbreviation***
Write Hit %	Percentage of write operations for which the addressed track was present in cache storage.	WRHITPCT

Appendix D. Information in Host Records

This appendix shows you which reports provide statistics for the various variables collected from the subsystems. Table D-1 lists the kinds of information shown in SAS reports and shows the associated internal variable names. Table D-2 on page D-4 shows the reports that provide data for each variable.

To use this appendix:

1. Find the statistic you want in the first column of Table D-1.
2. Obtain the corresponding variable name from the second column of the table.
3. Go to Table D-2 on page D-4 and find the variable name in the first column.
4. Read across the table. Wherever there is an x in the table, look at the top of the column to find a report that includes the statistic you want reported.

For example, to determine which reports indicate the amount of free space collected, look in the second column of Table D-1 and find “amount of free space collected (bytes).” The table indicates that the variable that contains this information is FSPCCOLL. Looking up that variable in Table D-2 on page D-4, you see an 'x' in the columns that indicate that you can find this statistic in any of the six kinds of summary device performance reports that show subsystem totals.

Table D-1 (Page 1 of 3). *SAS labels and their corresponding variables*

SAS label	**SAS variable**
% collected free space	FSPCCOL
% collected free space (Production)	FSPCCOLP
% collected free space (Test)	FSPCCOLT
% DASD fast writes directly toDASD	DFWCONST
% device active/capacity	IOCPRATE
% device connected to channel	PDVCONN
% device disconnected from channel	PDVDISC
% device was utilized (pdvconn+pdvdisc)	PDVUTIL
% drive module coeff of variation	COEFFVAR
% I/Os for which track was in cache	IOHIT
% of available reporting interval device	DEVAVAIL
% of reads for which track was in cache	READHIT
% of writes for which track was in cache	WRITHIT
% subsys active on channel	ACTVCHNL
% uncollected free space	FSPCUNC
% uncollected free space (Production)	FSPCUNCP
% uncollected free space (Test)	FSPCUNCT
# bytes read for free space coll (Production partition)	FSPCBTRP
# bytes read for free space coll (Test partition)	FSPCBTRT
amount of free space collected (bytes)	FSPCCOLL
ave % drive module utilization	DRVMODUT
ave % drive module busy	WGHTAVG

Table D-1 (Page 2 of 3). *SAS labels and their corresponding variables*

SAS label	SAS variable
ave num of reads per sec	READRATE
ave num of writes per second	WRITRATE
ave number I/Os per sec	IORATE
ave service time in millisecs	IOSTOTL
ave time device connect to channel msecs	IOSTCONN
ave time disconnected from channel msecs	IOSTDISC
cache storage in megabytes	RPTCACH
channel cluster	CLUSTER
channel I/Os per sec	CIORATE
channel interface identifier (A-P)	INTFID
channel interface name	INTFNAME
channel speed megabytes per second	CHANSPED
compress ratio	CMPRAT
cache size in megabytes	CUSTCACH
device address as known to the host	DEVNUM
device capacity	CAPACITY
device name	VDEVNAME
device number valid flag	DEVVALID
duration of collection interval	INTDUR
duration of reporting interval	INT
free min sectors avail for user (bytes)	FREBECAP
free back end sector (Production)	FREBESAP
free back end sector (Test)	FREBESAT
free space collection load (total)	SPCCOLL
free space collection load (Production partition)	SPCCOLLP
free space collection load (Test partition)	SPCCOLLT
highest % drive module util in 10% range	PCNT10
highest % drive module util in 20% range	PCNT20
highest % drive module util in 30% range	PCNT30
highest % drive module util in 40% range	PCNT40
highest % drive module util in 50% range	PCNT50
highest % drive module util in 60% range	PCNT60
highest % drive module util in 70% range	PCNT70
highest % drive module util in 80% range	PCNT80
highest % drive module util in 90% range	PCNT90
highest % drive module util in 100% range	PCNT100
hits per stage	HITSSTG
Iceberg device indicator	ICEBERG
input/output per sec per gig of func cap	ACCDEN
SVAA subsystem name	SUBSNAME
kilobytes per second	KBYTESEC

Table D-1 (Page 3 of 3). *SAS labels and their corresponding variables*

SAS label	SAS variable
low reference count	LOWREFCT
net capacity load % (total)	NCAPLDP
net capacity load % (Production)	NCAPLDPP
net capacity load % (Test)	NCAPLDPT
nonvolatile storage in megabytes	RPTNVS
NVS size in megabytes	NVSSIZE
offline cache in megabytes	OFFLCACH
offline cache in megabytes	RPTOFFL
part member for detail type recs	PARTM2
part member for part member type recs	PARTM1
partition membership	PARTMEMB
pinned cache in megabytes	PINDCACH
pinned cache in megabytes	RPTPIND
ratio of reads to writes	RWRATIO
reporting int/subsys summary var	INTSSN
reporting interval max date	DHIDATE
reporting interval max date	HIDATE
reporting interval max time	DHITIME
reporting interval max time	HITIME
reporting interval max date/time	HIGHTD
reporting interval max date/time	MAXENDTD
reporting interval min date/time	MINSTRTD
reporting interval start date	SRTDATE
stages per second	STGRATE
subsys interval summary variable	SSNINT
summary type	_TYPE_
tot num summarized recs	_FREQ_
total back-end capacity (Production partition)	TOTBECPD
total back-end capacity (Test partition)	TOTBECPT
tracks occupied per sec per cache mbyte	TRACKOCC
unit, tray, slot	UTS
virtual device identifier (0-4095)	VDID
volume serial number	VOLSER

Table D-2 (Page 1 of 4). *Reports containing the various output file variables. Each of the three kinds of report is divided into three columns, one for Device Performance reports, one for Cache Effectiveness reports, and one for both. Each of these three columns is further divided into two columns, one headed "D" and one headed "S"; the "D" column indicates a device-details report and the "S" column indicates a subsystem-totals report.*

Variable Name	INTERVAL SUMMARY						DAY/WEEK/MONTH SUMMARY						TIME-ORDERED SUMMARY					
	Dvce Perf		Cache Eff		Both		Dvce Perf		Cache Eff		Both		Dvce Perf		Cache Eff		Both	
	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D	S
FREQ		x				x		x				x		x				x
TYPE	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
ACCDEN	x	x			x	x	x	x			x	x	x	x			x	x
ACTVCHNL	x				x		x				x		x				x	
CAPACITY	x	x			x	x	x	x			x	x	x	x			x	x
CHANSPED	x				x		x				x		x				x	
CIORATE	x				x		x				x		x				x	
CLUSTER	x				x		x				x		x				x	
COEFFVAR		x				x		x				x		x				x
CUSTCACH			x	x	x	x			x	x	x	x			x	x	x	x
DEVAVAIL	x	x			x	x	x	x			x	x	x	x			x	x
DEVNUM	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
DEVVALID	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
DFWCONST			x	x	x	x			x	x	x	x			x	x	x	x
DHIDATE															x	x	x	x
DHITIME															x	x	x	x
DRVMODUT		x				x		x				x		x				x
FREBECAP		x				x		x				x		x				x
FSPCBTRP		x				x		x				x		x				x
FSPCBTRT		x				x		x				x		x				x
FSPCCOLL		x				x		x				x		x				x
IDATE	x	x	x	x	x	x												
HIGHTD													x	x	x	x	x	x
HITIME	x	x	x	x	x	x												
HITSSTG			x	x	x	x			x	x	x	x			x	x	x	x

Table D-2 (Page 2 of 4). *Reports containing the various output file variables. Each of the three kinds of report is divided into three columns, one for Device Performance reports, one for Cache Effectiveness reports, and one for both. Each of these three columns is further divided into two columns, one headed "D" and one headed "S"; the "D" column indicates a device-details report and the "S" column indicates a subsystem-totals report.*

Variable Name	INTERVAL SUMMARY						DAY/WEEK/MONTH SUMMARY						TIME-ORDERED SUMMARY					
	Dvce Perf		Cache Eff		Both		Dvce Perf		Cache Eff		Both		Dvce Perf		Cache Eff		Both	
	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D	S
ICEBERG			x	x	x	x			x	x	x	x			x	x	x	x
INT	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
INTDUR	x	x	x	x	x	x												
INTFID	x				x		x				x		x				x	
INTFNAME	x				x		x				x		x				x	
INTSSN		x	x	x	x	x		x	x	x	x	x						
IOCPRATE	x	x			x	x	x	x			x	x	x	x			x	x
IOHIT			x	x	x	x			x	x	x	x			x	x	x	x
IORATE	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
IOSTCONN	x	x			x	x	x	x			x	x	x	x			x	x
IOSTDISC	x	x			x	x	x	x			x	x	x	x			x	x
IOSTOTL	x	x			x	x	x	x			x	x	x	x			x	x
KBYTESEC	x	x			x	x	x	x			x	x	x	x			x	x
LOWREFCT			x	x	x	x			x	x	x	x			x	x	x	x
MAXENDTD							x	x	x	x	x	x						
MINSTRTD							x	x	x	x	x	x						
NCAPLDP		x				x		x				x		x				x
NCAPLDPP		x				x		x				x		x				x
NCAPLDPT		x				x		x				x		x				x
NVSSIZE			x	x	x	x			x	x	x	x			x	x	x	x
OFFLCACH			x	x	x	x									x	x	x	x
PARTMEMB	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
PARTM1							x	x	x	x	x	x	x	x	x	x	x	x
PARTM2							x	x	x	x	x	x	x	x	x	x	x	x
PCNT10	x				x		x				x		x				x	

Table D-2 (Page 3 of 4). *Reports containing the various output file variables. Each of the three kinds of report is divided into three columns, one for Device Performance reports, one for Cache Effectiveness reports, and one for both. Each of these three columns is further divided into two columns, one headed "D" and one headed "S"; the "D" column indicates a device-details report and the "S" column indicates a subsystem-totals report.*

Variable Name	INTERVAL SUMMARY						DAY/WEEK/MONTH SUMMARY						TIME-ORDERED SUMMARY					
	Dvce Perf		Cache Eff		Both		Dvce Perf		Cache Eff		Both		Dvce Perf		Cache Eff		Both	
	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D	S
PCNT100	x				x		x				x		x				x	
PCNT20	x				x		x				x		x				x	
PCNT30	x				x		x				x		x				x	
PCNT40	x				x		x				x		x				x	
PCNT50	x				x		x				x		x				x	
PCNT60	x				x		x				x		x				x	
PCNT70	x				x		x				x		x				x	
PCNT80	x				x		x				x		x				x	
PCNT90	x				x		x				x	x	x				x	
PDVCONN	x	x			x		x	x			x	x	x	x			x	x
PDVDISC	x	x			x	x	x	x			x	x	x	x			x	x
PDVUTIL	x	x			x	x	x	x			x	x	x	x			x	x
PINDCACH			x	x	x	x									x	x	x	x
READHIT			x	x	x	x			x	x	x	x			x	x	x	x
READRATE			x	x	x	x			x	x	x	x			x	x	x	x
RPTCACH			x	x	x	x			x	x	x	x			x	x	x	x
RPTNVS			x	x	x	x			x	x	x	x			x	x	x	x
RPTOFFL			x	x	x	x									x	x	x	x
RPTPIND			x	x	x	x									x	x	x	x
RWRATIO			x	x	x	x			x	x	x	x			x	x	x	x
SPCCOLL		x				x		x				x		x				x
SPCCOLLP		x				x		x				x		x				x
SPCCOLLT		x				x		x				x		x				x
SRTDATE							x	x	x	x	x	x						
SSNINT														x				x

Table D-2 (Page 4 of 4). *Reports containing the various output file variables. Each of the three kinds of report is divided into three columns, one for Device Performance reports, one for Cache Effectiveness reports, and one for both. Each of these three columns is further divided into two columns, one headed "D" and one headed "S"; the "D" column indicates a device-details report and the "S" column indicates a subsystem-totals report.*

Variable Name	INTERVAL SUMMARY						DAY/WEEK/MONTH SUMMARY						TIME-ORDERED SUMMARY					
	Dvce Perf		Cache Eff		Both		Dvce Perf		Cache Eff		Both		Dvce Perf		Cache Eff		Both	
	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D	S
STGRATE			x	x	x	x			x	x	x	x			x	x	x	x
SUBSNAME	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
TOTBECPP		x				x		x				x		x				x
TOTBECPT		x				x		x				x		x				x
TRACKOCC[1]			x	x	x	x			x	x	x	x			x	x	x	x
UTS		x				x		x				x		x				x
VDEVNAME	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
VDID	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
VOLSER	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
WGHTAVG		x				x		x				x		x				x
WRITHIT			x	x	x	x			x	x	x	x			x	x	x	x
WRITRATE			x	x	x	x			x	x	x	x			x	x	x	x

[1] Only at the subsystem level (not on device details)

Appendix E. Using Data Extraction

This appendix provides suggestions for creating graphs of the extracted data in various operating environments, and defines the variables that may be specified for data extraction (see “Extracting Summarized History Data” on page 7-50).

Overview of Data Extraction

The data-extraction function of Reporter reduces the amount of data involved in a report so that the data can be used to produce a meaningful graphic report—either with SAS/GRAPH in an OS/390 environment, on a UNIX workstation or a PC, or with an application that can process comma-delimited files in any of these environments or on a Macintosh personal computer. You can also process the data with SAS or with any other personal computer application that takes comma-delimited output.

The general process is:

1. Use the SVAA panels to create a SAS extraction program (see “Extracting Summarized History Data” on page 7-50). SVAA uses data from the history files you specify as input.
2. Execute the SAS extraction program.
   - This happens automatically, using JCL created by SVAA (see “Executing the Program to Produce Comma-Delimited Output” on page A-7).
3. The SAS extraction program produces output in one of the following forms, as you specify on the panels:
   - Comma-delimited output file. Process this file either on a workstation or a personal computer, using any program that takes comma-delimited data as input.
   - SAS output file. Process this file with SAS/GRAPH on a mainframe. The SVAA SIBSAMP data set includes sample programs that produce the seven kinds of graphs that you can specify on the panels. You must modify the SAS LIBNAME statement in these programs as necessary.
   - SAS transport file. Download this file to a workstation or to a personal computer and process it with SAS and SAS/GRAPH.

Later sections of this appendix provide additional information about step 3 in various environments.

Preallocating Output Files

Before using the SVAA panels to create a SAS extraction program, you must preallocate the output data sets, as follows:

- Comma-delimited output files:
  DCB=(RECFM=VB,LRECL=4092,BLKSIZE=*n*)
  where *n* is at least 4096.
- SAS output files:
  DCB=(RECFM=FS,LRECL=6144,BLKSIZE=6144)
- SAS transport files:
  DCB=(RECFM=FB,LRECL=80,BLKSIZE=*n*)

where *n* is a multiple of 80.

Creating the SAS Extraction Program

You use the SVAA panels as described in "Extracting Summarized History Data" on page 7-50 to create a SAS program to extract the data from a current history file for graphing or other processing.

The examples in this appendix assume that you have run the SAS program to extract the data and that the output program is stored in a file called *hlq*.**EXTRACT.SAMPLE** in OS/390.

There is a SAS transport file called, perhaps, *hlq*.**EXTRACT.TRANSPORT**. This file can be transferred to any OS/390 or UNIX system, or to a personal computer.

Note: In the examples in this appendix, *hlq* is your installation-defined higher-level node or nodes.

Running the Output Program

The data extraction function of SVAA produces a SAS program that you can run to produce output files that can be used in various environments to produce graphs of the extracted data. The output files can be comma-delimited or in SAS format.

Extracting the Data

Run the output SAS extraction program in OS/390.

Downloading SIBSAMP Members to the Workstation

For creating the graphs, SVAA includes sample programs both in the SIBSAMP data set and on a floppy disk. Use the following process to download the sample programs from SIBSAMP.

The sample program files must be transferred to the workstation with EBCDIC-to-ASCII translation, because all the data in the files are in character format.

To transfer a file from OS/390 to a workstation using the File Transfer Program (FTP) on the mainframe:

1. Enter **FTP** *workstation-name*, where *workstation-name* is the defined name or TCP/IP address of your workstation.
2. When prompted, enter your workstation user ID and password.
3. Enter **ASCII** to make sure that EBCDIC-to-ASCII translation occurs.
4. To transfer each of the files from OS/390, enter: **PUT** *'hlq*.**SIBSAS (SIBSGR***nn***)' /***path***/SIBSGR***nn***, where *nn* is **01, 02, ..., 07** in turn.

Creating the Graphs

If you created a SAS output program, run the sample SAS programs SIBSGR01 - SIBSGR07 to produce the graphs. You can run these programs in either OS/390 or UNIX.

Downloading Comma-Delimited Files

Before the extracted data can be processed on the workstation, the files must be downloaded. The process to follow depends on the kind of file.

Comma-delimited files must be transferred to the workstation with EBCDIC-to-ASCII translation, because all the data in the files is in character format, consisting of fields that are in quotes and separated by commas.

To transfer a comma-delimited file from a mainframe computer to a workstation using the File Transfer Program (FTP) on the mainframe:

1. Enter **FTP** *workstation-name*, where *workstation-name* is the defined name or TCP/IP address of your workstation.
2. When prompted, enter your workstation user ID and password.
3. Enter **ASCII** to make sure that EBCDIC-to-ASCII translation occurs.
4. When transferring from OS/390, enter: **PUT** *'hlq.***EXTRACT.SAMPLE'** *workstation-name*.

   Note: *workstation-name* must follow the workstation's file-naming convention and must include the full path name.

Downloading SAS Files

SAS files (either a SAS extracted-data file or a SAS transport file) must be transferred to a workstation without EBCDIC-to-ASCII translation, because data in these files is in mixed binary and character formats. SAS translates the character fields as necessary after the data has been downloaded.

To transfer a SAS file from a mainframe to a workstation using FTP on the workstation:

1. Enter **FTP** *hostname*, where *hostname* is the defined name or TCP/IP address of the mainframe.
2. When prompted, enter your user ID and password for your logon to the mainframe.
3. Enter **BINARY** to make sure that EBCDIC-to-ASCII translation does not occur.
4. To download the files in OS/390, enter: **GET** *'hlq.***EXTRACT.SAMPLE'** *workstation-filename*.

Notes:

1. *workstation* must follow the workstation's file-naming convention and must include the full path name.
2. Once the SAS file has been downloaded to the workstation, you can access the device-detail and subsystem-totals output as separate SAS library members. (See the sample SAS/GRAPH programs distributed in the SIBSAMP data set for examples of the required FILENAME statements.) If you created SAS extract files as well as SAS transport files, the SAS member names of the exported output are the same as the member names specified for the SAS file members. If you did not create SAS extract files, the member name is TEMPDET for device totals and TEMPTOT for subsystem totals.

Graphing Extracted Data on a PC

Data in a comma-delimited extract file can be processed with any application that accepts comma-delimited input. One way is with an Excel macro, a sample of which can be found in the SIBSAMP library distributed with SVAA.

Downloading the Sample Excel Macro

There is a sample Excel macro (in binary form) named SIBXGR00 in SIBSAMP and also on a diskette supplied by StorageTek. To transfer the macro from the mainframe to a personal computer using FTP on the mainframe:

1. After entering FTP and setting up the connection, enter **BINARY**.
2. In OS/390, enter: **PUT '*hlq*.SIBSAMP(SIBXGR00)' C:\EXCEL\SIBXGR00.XLM**.

Running the Excel Macro

To run the sample Excel macro:

1. Open Excel.
2. Select **File** from the menu bar.
3. Select **Open** from the file menu.
4. Select **SIBXGR00.XLM** to load the macro (the macro has the hidden attribute, so it does not appear on your display).
5. In Excel 4:
   a. Select **Macro** from the menu bar.
   b. Select **Run**.
   c. Highlight **SIBXGR00.XLM** and click on **OK**.
6. In Excel 5:
   a. Select **Tools** from the menu bar.
   b. Select **Macro**.
   c. Highlight **SIBXGR00.XLM** and click on **Run**.
7. A dialog box with the title “Select Worksheet to Chart” appears. The upper-left corner of the dialog box contains a list of the worksheets that you have already opened. If a worksheet containing the data you want to chart has already been opened, you can select it from this list (only one worksheet name appears at first—you can click the down arrow to see the names of the rest of the open worksheets, and highlight the one you want). Then click on **OK**.
8. If the worksheet containing the data you want to chart has not yet been opened, click on the “Open New File...” box in the lower left corner of the dialog box (note that this option is not available on the Macintosh, where the worksheet must have been opened previously).
9. A new dialog box called “Open New worksheet” now appears. Select the desired file from the list of files in the current directory. If the file you want is not in the current directory, select the appropriate directory from the list of directories, then select the desired file. If the file is still in comma-delimited format (that is, it has not yet been converted to Excel format), click on the “comma-delimited” option in the lower-left corner of this dialog box. (Excel reads in the file as a comma-delimited file and saves it as an Excel file with an .XLS extension in the Windows environment.

10. The SIBXGR00.XLM macro then opens the file. Note that if the selected file had previously been opened, the macro reverts the file to the copy saved on the disk. Therefore, be careful not to use this option if you have made changes to the worksheet that have not yet been saved.

11. Once you have selected an already open worksheet or you have opened a new one, a new dialog box called "Select Graphs" appears. Only those graphs for which the worksheet has data are available for selection (the rest are dimmed). If the worksheet was created from extracted subsystem totals data, the "I/O service time and functional device utilization" graph is not available. If the worksheet was created from extracted device totals data, the "I/O service time and functional device utilization graph" is the *only* graph available and is automatically selected, if all the required fields are present. Select the desired graphs from the list and click on OK. The graphs are generated and appear in separate windows. Note that the macro may take several minutes to produce a graph, depending on the amount of data present. (This is especially true for the device-totals graph.)

    Note: If you select too many graphs, or if you are graphing a large amount of data, you may run out of Windows® resources. In that case, close any graphs you have already produced and run the macro again, this time selecting fewer graphs.

Opening Comma-Delimited Files

To open a comma-delimited extract file with Excel, rather than using the SIBXGR00.XLM macro, follow these steps:

1. Select **File** from the Excel menu bar.
2. Select **Open** from the File menu.
3. With Excel 4, click on the **Text** button. Then select the **comma** option and click on OK.
4. Highlight the desired file or files, then click on OK.
5. To save the file as an Excel file, select **File** from the menu bar. Select **Save As...** from the File menu. Under **Save as File Type**, select "Normal." In Windows, the file is then saved with the .XLS extension.

LIBNAME statements in SAS

The data extract function creates a SAS program that produces the output files in comma-delimited or SAS format. SIBSAMP contains sample SAS-format files named SIBSGR*nn*, where *nn* is **01** through **07** (these numbers correspond to the seven types of graphs you can create—see page 7-57). To run SAS/GRAPH using the sample programs, you must provide LIBNAME statements and you must modify the SET statement to point to the correct libdef and filename. The format of these statements depends on where the data extraction program was run and where you are running SAS. The following examples refer to output files named *hlq*.EXTRACT.SAMPLE with SAS member name EXTRACT in OS/390.

Produced on OS/390, Running on OS/390

No LIBNAME statement is needed.

The format of the SET statement is:

SET SIBSGR*nn*.*membername*

where **SIBSGR*nn*** points to the ddname in the JCL that points to the extracted data set and *membername* is the SAS member name.

Example: **SET SIBSGR01.EXTRACT;**

Produced on OS/390, SAS-Transport File Running on UNIX

The format of the LIBNAME statement is:

LIBNAME SIBSGR*nn* XPORT '/path/filename';

Example: **LIBNAME SIBSGR01 XPORT '/a/b/EXTRACT.SAMPLE';**

The format of the SET statement is:

SET SIBSGR*nn*.*membername*;

where *membername* is the SAS member name.

Example: **SET SIBSGR01.EXTRACT;**

Miscellaneous Information about Data Extraction

In addition to the above, data extract users should be aware of the following items.

Estimated Work Space

SVAA calculates the number of bytes required to store a data extract output file as 8 times the number of unique variables selected, times the number of days specified, times the number of intervals within the time range.

Saving a SAS Graph in a Postscript® File

The sample SIBSGR*nn* files in SIBSAMP allow you to save the output in a Postscript file instead of displaying it on the screen. To create a Postscript file for printing:

1. Edit the SIBSGR*nn* file from SIBSAMP.
2. Locate the phrase “Remove asterisk to Save Output.”
3. Remove the asterisk from column 1 of that line.

To change back to displaying the output on the screen, replace the asterisk.

Excel Graphs

You can produce all seven graphs with SIBGRX00.XLM. You select the graphs you want when you run the macro, providing you had specified those graphs on the panels.

The macro performs some calculations to produce the graphs. Therefore, these charts are not automatically updated when the underlying spreadsheet changes; you must regenerate these charts by running the macro again.

Volsers in Extracted Output

If a functional device is offline or not defined to the collecting host during data collection, the volume serial number is unknown. In that case, the volume serial number field in the data extract output contains either the user-defined name of the functional device, followed by a single asterisk or, if the functional device name is blank, the functional device ID, followed by two asterisks.

Data Extraction Variables

This section lists and describes the reporting variables that can be specified for data extraction.

Data for some variables is not extracted from the source files under certain circumstances, because doing so would create redundant data for functional devices when the data was collected over a number of channels. In this case, SVAA extracts only channel totals. Table E-1 shows the two groups of variables that are NOT extracted and the circumstances under which this happens. The two groups are:

Set A *chansped, cluster, intencur, intfid, intfname*

Set B *capacity, devnum, devvalid, vdevname, vdid, volser*

The first column of the table shows the type of source file from which the data is to be extracted. The second column shows the variables that are excluded from the device summary file; the third column shows the variables that are excluded from the subsystem totals summary.

Table E-1. *Variables NOT extracted for each summary type*

Source	Summary type requested	
	Device	**Subsystem**
Device detail file		For the functional devices, Set A plus Set B
Detail detail and channel files	Set A	For both functional devices and channels, Set A plus Set B
Channel file		For the channels, Set A plus Set B

The variables that may be defined for data extraction are: [1]

accden Access density

actvchnl Percent subsystem active on channel

availtim Device available time in milliseconds

busytim Total drive-module-busy time

busytime Total drive module utilization in milliseconds

bypcareq Number of bypass cache requests

bytetrak Bytes per track

cacsegct Number of device cache utilization segments

cadasdxf Cache to DASD transfers (destages)

capacity Functional capacity in gigabytes

caseqxfr Number of sequential stages (prestages)

caxfrstg Normal stages

[1] For non-SVA devices, SVAA provides only cache-effectiveness data. Therefore, some of the following variables will have zero values in the extracted data file.

chansped Channel speed
chantype Channel interface type
ciorate Channel I/O rate
cluster Cluster ID
coeffvr Drive module utilization coefficient of variation
collid Collection ID
conntime Total time device is connected in milliseconds
cpuid ID of CPU on which data collection was run
custcach Cache size in megabytes
devactiv Number of end-of-chain events
devavail Percentage functional device available
devnum Device number as known to operating system
devutlti Total time device is utilized in milliseconds
devvalid Device number is valid indicator
dfwnorht DASD Fast write normal hits
dfwnorwr DASD Fast write normal write requests
dfwnvsct DASD Fast write constraint count
dfwseqht DASD Fast write sequential hits
dfwseqwr DASD Fast write write requests
drvmdtim Interval duration in milliseconds
drvmdtm Total drive module available time
drvmdut Average percentage drive module utilization
enddate Interval ending date
endtime Interval ending time
fastwrts Fast-write requests
febytxfr Number of front-end read bytes transferred
febytxfw Number of front-end write bytes transferred
frdealct Total number of frames deallocated
frebecp Free (collected plus uncollected) back-end capacity in gigabytes
frebesc Total (collected plus uncollected) back-end capacity in gigabytes
fspcbtr Bytes read for free-space collection
fspccol Total amount of free-space collected
funccap Total functional capacity in gigabytes
hitsstg Hits per stage
iceberg Inhibit cache-loading requests
inhcalrq Number of requests to inhibit cache-loading
intencur Channel enabled

intfid	Interface ID
intfname	User-assigned interface name
iohit	I/O hit percentage
iorate	Average I/O rate (device activity)
ioratec	I/O rate (cacheable I/Os)
iostconn	Average connect time per I/O
iostdisc	Average disconnect time per I/O
iostotl	Average service time per I/O
kbperio	Kilobytes transferred per I/O
kbytesec	Average kilobytes transferred per second
lowrefct	Low reference deallocation percentage
lowrfcnt	Frames deallocated with fewer than two references
maxconn	I/O connect time at maximum I/O rate
maxdisc	I/O disconnect time at maximum I/O rate
maxiort	Maximum subsystem I/O rate (device activity)
maxiost	I/O service time at maximum I/O rate
maxkbsc	Maximum kilobytes per second transferred
mincols	Minimum collected free space in gigabytes
ncaplda	Net capacity load amount in gigabytes
ncapldp	Net capacity load percentage
numberio	Number of end-of-chain events at channel interface
nvssize	Nonvolatile storage (NVS) size
offlcach	Offline cache
offread	Reads when device was not cached
offwrite	Number of bytes when device was not cached
partmemb	Partition membership
pcolfsp	Percentage collected free space
pdfwcnst	DFW constraint percentage
pdvconn	Average functional device connected percentage
pdvdisc	Average functional device disconnected percentage
pdvutil	Average functional device utilization percentage
pindcach	Pinned cache
prfsegct	Number of device performance segments
primcapa	Primary capacity in cylinders
puncfsp	Percentage of uncollected free space
rdcfwhit	Read cache fast-write hits
rdcfwreq	Read cache fast-write requests

rdnorhit	Read normal hits
rdnorreq	Read normal requests
rdseqhit	Read sequential hits
rdseqreq	Read sequential requests
readbyts	Bytes read from drive module
readhit	Read hit percentage
readhits	Read hits
readrate	Read rate
readreqs	Read requests
rgrpcnt	Number of arrays
rwratio	Read/write ratio
scsilbsz	SCSI logical block size
scsilbt	SCSI logical blocks per track
smfdate	Date record presented to SMF
smfflg	SMF system indicator
smfrty	SMF record type
smfsid	SMF system ID
smfssi	OS/390 subsystem ID
smftime	Time record presented to SMF
spccoll	Free-space collection load
stages	Stages
stgrate	Stage rate
subdurat	Interval duration in milliseconds
subsname	Subsystem name
svcperkb	Service time per kilobyte transferred
ticobuch	Time control unit busy at channel in milliseconds
totbecp	Total back-end capacity in gigabytes
trackcyl	Number of tracks per cylinder
vdevname	Functional device name assigned by user
vdevtype	Device type
vdid	Functional device ID
version	Version of this record
volser	Volume serial number
voltype	Volume type
wrcfwhit	Write cache fast-write hits
wrcfwreq	Write cache fast-write requests
wrtebyts	Bytes written to drive module

writhit	Write hit percentage
writhits	Write hits
writrate	Write rate
writreqs	Write requests
wrnorreq	Write normal requests
wrseqreq	Write sequential requests

****

Appendix F. Point-in-Time Reports

Overview

The SVAA point-in-time reports provide detailed statistics of SVA subsystem functional device or I/O interface conditions at a given point in time. They are produced only through the SVAA Web-based interface (WBI). The reports cannot be produced through the CLI (command line interface), and SAS is not a requirement for producing these reports. Finally, the point-in-time reports cannot be produced using SVAA for VM since SVAA for VM does not communicate with the SVAA server.

This appendix explains how to generate and interpret the reports.

Note: The point-in-time reports' online Help also contains some of this information.

There are three point-in-time reports:

- Subsystem Report: Summarizes the specified SVA subsystem, including cache size and efficiencies, read/write efficiencies, and defined device types and compression ratios.
- Device Report: Reports DTL (domain, target, LUN), FDID, address, type, average usage, etc., for each specified functional device.

  Note: Currently this report offers a functional view of the SVA subsystem only; it does not offer a logical view (e.g., the functional devices that make up a SCSI larger LUN device).
- I/O Interface Report: Reports interface name, type, activity level, and efficiencies for each specified I/O interface.

The point-in-time reports present a picture of the SVA as of a specific point in time. The report values come from the SVA subsystem at the time the report is generated. The values shown on a point-in-time report may not exactly match comparable values displayed in SVAA displays and queries or the SAS reports.

Setting up the Web-Based Interface (WBI)

A user must be able to access the SVAA server through the WBI in order to run the point-in-time reports. The WBI provides a graphical user interface (GUI) to the SVAA server. There can be up to five (5) connections to the Web browser.

This section instructs SVAA administrators on how to set up and maintain the SVAA server (WBI). These functions require administrative privileges.

Note: The SVAA Server must be shut down and restarted before any changes to the WBI will take effect. (See the STOP SFC and START SFC subcommands for details.) Changes made using the SET WEBPORT subcommand require that the SVAA Server be recycled.

Enabling the WBI

You initially enable the WBI as part of SVAA Server initialization by specifying the SET WEBPORT subcommand in the SIBSFCxx parmlib member. See "Step 9c: SIBSFCxx Member" in Chapter 3 of the SVAA for OS/390 Installation, Customization, and Maintenance manual.

When enabling the WBI, you assign a WBI port number to the SVAA Server. The WBI port is the TCP/IP port used for communications between the WBI and SVAA Server. Enabling the WBI permits users to access the SVAA Server through Web browsers connected to this port. Once enabled, the WBI remains so until explicitly disabled.

Changing the WBI Port Number

Changing the WBI port number for an SVAA server overwrites the current port number. The new port number remains in effect until explicitly changed.

Specify the new WBI port number by modifying the SIBSFCxx parmlib member's SET WEBPORT subcommand.

Disabling the WBI

Disabling the WBI prevents all users from displaying point-in-time reports through a Web browser. The WBI remains disabled until explicitly re-enabled.

To disable the WBI, specify SET WEBPORT(OFF) in the SIBSFCxx parmlib member.

Re-enabling the WBI

Re-enabling the WBI makes it available to all users for displaying point-in-time reports. The WBI remains enabled until explicitly disabled.

Use the SET WEBPORT subcommand and specify a port number.

Security Considerations

The point-in-time reports generate only query-type requests to SVAA; therefore, the reports require no special privileges. All users are allowed to generate all point-in-time reports

Generating the Reports

This section instructs users on how to generate point-in-time reports through the Web-based interface (WBI).

Supported Browsers

The point-in-time reports are generated from a Web browser connected to the SVAA WBI. The WBI supports all of the following browsers:

- Microsoft Internet Explorer (IE) 5.5 and 6.0
- Netscape R4.7.x, R6.1, and R6.2.

Accessing the SVAA Server

Use the following procedure to use a Web browser to access the SVAA server.

1. Open a browser window. (See “Supported Browsers” above for browsers you can use.)
2. In the Address/Location field of the browser window, enter the following URL:

   http://host_name:port_number

   where:

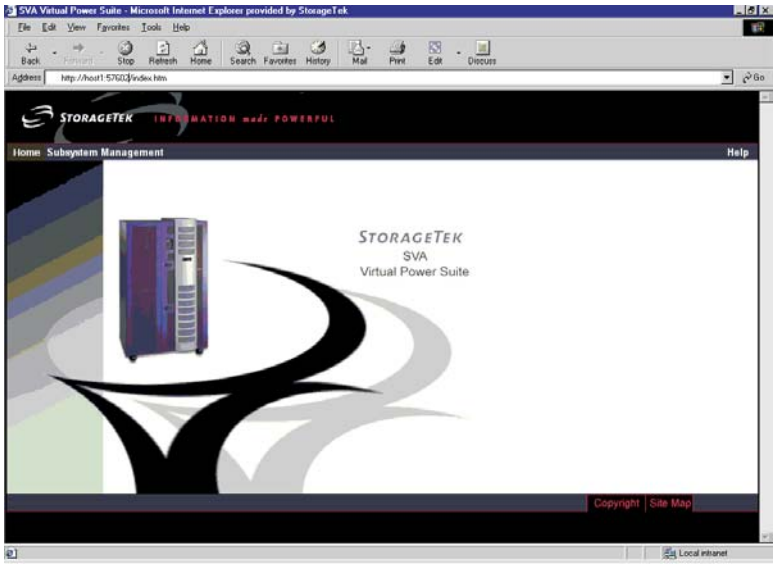
- *host_name* is the name of the host running the SVAA server you wish to access.
- *port_number* is the defined WBI port of the SVAA server.

  Note: Your SVAA administrator can provide you with this information.

For example, to access an SVAA server running on a host named "host1" on port 57602, enter the following:

`http://host1:57602`

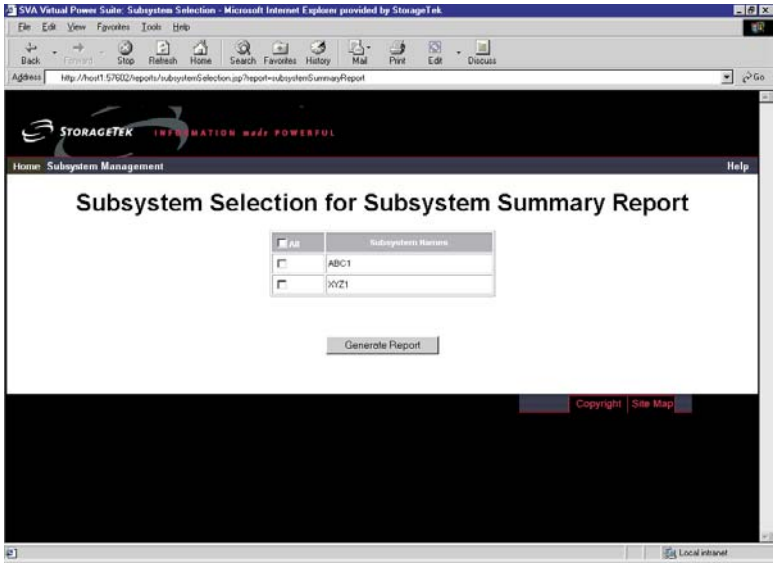
3. The SVA Virtual Power Suite screen appears.



Generating a Report

Use the following procedure to generate a point-in-time report for one or more SVA subsystems.

1. From the SVA Virtual Power Suite screen, click **`Subsystem Management`**.
2. Click **`Reports`**.
3. Click the report you wish to generate. Current selections are:
   - Subsystem Report
   - Device Report
   - I/O Interface Report
4. The Subsystem Selection screen appears. The screen displays all SVA subsystems to which the SVAA server currently has access. All point-in-time reports use this screen.



5. Specify the SVA subsystem(s) to be included in the report:
   - Click **ALL** to include all SVA subsystems.
   - Click individual subsystem names to only include those subsystems.

     Note: If the SVAA Server currently does not have access to any SVA subsystems, you will see a message indicating that no subsystems are available.
6. Click **Generate Report**.
7. The report is generated and displayed in the browser window.

Printing Reports

You can print a point-in-time report by using the standard file print features available from your browser. See the documentation for your specific browser for detailed instructions. Typically, you would select **File>Print** from the browser pull-down menu.

Saving Reports

You can save a point-in-time report by using the standard file save features available from your browser. See the documentation for your specific browser for detailed instructions. Typically, you would select **File>Save As** from the browser pull-down menu and assign a location and name to the report.

Note: Be sure to give the file an **.htm** or **.html** extension to identify it as an HTML file.

Interpreting the Reports

For detailed information about each field on the report, use the point-in-time report's Online Help feature.

Glossary

This glossary is included in each book in the Shared Virtual Array Administrator library. All of the terms are associated with SVAA, but not all are used in this specific document.

A

Alias. A pseudo-device used by the operating system to support an additional I/O path to a Base device. Each Alias device supports one additional I/O to a Base. See also: Base and Parallel Access Volume.

array. A group of storage devices that are used collectively to achieve data redundancy and/or improved performance. In the SVA, an array consists of either 7 or 15 drive modules. See also: dual-redundancy array.

array cylinder. The collection of all physical cylinders in a dual-redundancy array that have the same physical cylinder address (CC). The SVA allocates back-end space in units of array cylinders. There are two types of array cylinders: free and allocated.

array device. The disk devices that are logically grouped together when a FORM ARRAY command is issued at the local operator panel or from SVAA.

array track. The collection of all physical tracks in a dual-redundancy array that have the same physical track address (CC, HH).

B

back-end storage. The data storage portion of a storage subsystem. In the SVA, the disk arrays.

Base. A real device that supports additional I/O paths to itself in the form of Alias devices. Each Alias device supports one additional I/O to a Base. Multiple Alias devices can be associated with a single Base. See also: Alias and Parallel Access Volume.

base functional device ID (BFDID). The functional device identifier that maps to or from the lowest (base) interface address on a given channel.

C

cache. Solid state, random access memory that is located in a controller. The cache retains frequently used data for faster access by the channel. In the SVA, all data access is through cache.

cache fast write (CFW). A form of fast write in which data is written directly to cache storage without using nonvolatile storage and is available for later destaging.

Capacity on Demand (COD). A feature that allows SVA customers to exceed their purchased physical capacity (PCAP) limit by up to 860GB of additional temporary effective capacity to prevent writes from being blocked during temporary and sometimes critical usage periods.

channel end. The indication from the channel that it has completed an operation.

channel interface. The Disk Array Controller circuitry that attaches to the host channels.

cluster. See storage cluster.

collected free space %. The percentage of array cylinders that are free array cylinders (collected and completely free of user data).

compaction. The SVA process that eliminates inter-record gaps normally associated with CKD DASD. Compaction reduces the amount of wasted disk array space, thus reducing the net capacity load on the subsystem.

compression. The SVA process that reduces the size of data records by translating them to a different encoding scheme that requires fewer bytes of real storage.

controller. See Disk Array Controller.

count-key-data (CKD). A recording format that writes variable-length records. Each record consists of 1) a count field, which specifies the length of the (optional) key field and data field of the record, 2) the (optional) key field, and 3) a data field. The first record on each track contains a fourth field, home address.

current data. User data, stored in a disk array, that has valid pointers from internal SVA mapping tables.

D

DASD fast write (DFW). A form of fast write to cache in which data is written concurrently to cache and nonvolatile storage (NVS) and is subsequently scheduled for destaging to the disk arrays. Both copies are retained in the SVA Disk Array Controller until the data is completely written to the disk arrays.

Data Bridge. A pair of devices used by Power PPRC to transmit all tracks on all primary devices from the primary subsystem to the secondary subsystem. These devices are not used to store customer data.

dedicated connection. In an Enterprise Systems Connection Director (ESCD), a connection between two ports that is not affected by information contained in link frames. This connection restricts these ports from communicating with any other port. The two ports have a dedicated connection that appears as one continuous link.

Deleted Data Space Release (DDSR). An SVAA facility for OS/390 that informs the Disk Array Controller when functional volume data sets are deleted. The physical disk array space occupied by the deleted data can immediately become free space, thereby reducing the net capacity load on the subsystem.

destage. The nonsynchronous write of new or updated data from the cache storage or nonvolatile storage to the Disk Array Units.

device. See (1) drive module and (2) functional device.

device end. An indication from an I/O device that it has ended an operation.

device reconstruction. The SVA automatic background function of recreating and rewriting all of the data that was stored on a failed device to a spare device using the functional track recovery process.

direct access storage device (DASD). A storage device in which the medium is always available to the read/write head without having to be mounted by an external agent.

disk array. The SVA's logical grouping of drive modules. See also: dual-redundancy disk array.

disk array capacity. The formatted physical capacity of a disk array excluding redundancy data.

Disk Array Controller. The SVA control unit that provides the interface intelligence between the host(s) and the back-end storage.

Disk Array Unit (DAU). A single physical frame containing drive modules that comprise the disk array storage in an SVA subsystem.

domain. See SCSI domain.

drain. The SVA process that gradually moves data stored on a device or a disk array to other devices. Drain operations allow for the nondisruptive deinstallation of a device or a Disk Array Unit.

drive module. A disk storage device consisting of the access arms and heads, disk surfaces, and the supporting electronics required to locate, write, and read data. Each drive module is physically packaged as a single field-replaceable unit (FRU) within the SVA.

drive reconstruction. See device reconstruction.

dual-redundancy disk array. A disk array that allows for real-time automatic recovery of data from up to two failed devices within the array.

In the V2X and V960 SVAs, a dual-redundancy disk array consists of 15 (13+2) drive modules. The array has a capacity equivalent to 13 drives of user data and 2 drives of redundancy data. (In the SVA, redundancy data is distributed among all 15 drives).

In the 9500 and earlier SVAs, arrays of 7 (5+2) drive modules can also be formed.

Dynamic Configuration. An SVA feature that allows the channel interfaces and up to 4096 functional volumes to be defined and/or altered. The functional configuration of an SVA subsystem can be determined by user requirements rather than available drive modules.

E

ECAM device. A functional device over which SVAA-based communication between the SVA Disk Array Controller and the host CPU(s) takes place.

esoteric names. The names a user assigns to DASD volumes that have the same device type.

ESCON channel. A channel that uses ESCON cables to transmit data between the host and the Disk Array Controller.

Extended Control and Monitoring (ECAM). The communications protocol that permits communication between SVAA and the SVA.

extent. A range of disk addresses expressed as a cylinder head range (CCHH) for a CKD device, or a logical block address (LBA) for a SCSI device.

F

fast write. A write operation that does not require immediate synchronous transfer of data to a DASD device, thus reducing the time an application must wait for channel end and device end for an I/O operation.

fault symptom code (FSC). An error code, generated by a control unit or subsystem, that points to the area or FRU most likely causing a problem.

fault tolerance. The capability of a subsystem to continue operating without interruption and/or intervention despite a failure within the subsystem (e.g., hardware, power, cooling). Fault tolerance is generally measured in relation to inherent reliability, availability, serviceability, and recoverability for the product.

FDID map. See functional device identifier mapping.

fence. The automatic or manual separation of a logical path or physical component from the remaining operating portion of the subsystem. The fencing process provides for continuous operation of the subsystem and allows for deferred nondisruptive servicing of field-replaceable units (FRUs) via hot-plugging.

A logical barrier on a node or path that prevents the use of that node or path.

FICON channel. A channel that uses fiber connections to transmit data between the host and the Disk Array Controller.

field-replaceable unit (FRU). The smallest self-contained component that can be individually replaced during a service or repair action.

fixed block architecture (FBA). (Contrast with CKD) A recording format in which every track of the device is formatted with a fixed number of fixed-length records (generally called sectors), each of which contains an identifier (ID) field and a data field.

flexvolume. A 3380 or 3390 CKD volume defined with less than the maximum number of cylinders. The range of cylinders allowed depends on the device type.

free array cylinder. An array cylinder that contains no current or non-current user data.

free space collection (FSC). The automatic SVA background task that relocates data from fragmented array cylinders in order to collect free space into empty array cylinders. Free space collection maximizes the efficiency of array cylinder writes.

free space collection load. The average percentage of array cylinder space that must be relocated in order to create empty array cylinders in the SVA.

front end. The portion of the SVA Disk Array Controller data path that passes data between the channels and the cache.

functional. The term used to describe the SVA interface as viewed by the host, application, and users. This interface appears as a 3990-3 subsystem interface.

functional/allocated. The user-allocated portion of a functional volume's space; that is, data sets as defined in the VTOC.

functional capacity. The data storage capacity that the host, application, and users view. Used in reference to the space available for storing data in (1) a single functional device, or (2) all defined functional devices in an SVA subsystem.

functional device. The volume image that the host operating system receives when the “Read Device Characteristics” CCW is issued.

functional device identifier (FDID). The identifier for a functional device as it is known to the SVA. FDIDs range from 0 to FFF (hexadecimal) or from 0 to 4095 (decimal).

functional free space. The unallocated/unused portion of a functional volume's space, as defined in the VTOC.

functional track. The equivalent of a 3380- or 3390-DASD track. A functional track record is stored on contiguous sectors in an allocated array cylinder.

functional track directory (FTD). The SVA internal mapping table that contains one entry for each functional track associated with the functional volumes currently defined by the user.

functional track recovery (FTR). The automatic SVA process of recovering data from a physical track that is unreadable due to a media defect or a failed device. The SVA accomplishes functional track recovery by reading and processing the user data and redundancy data at corresponding physical track locations on the remaining devices in the array.

functional volume. See functional device.

G

generation data group (GDG). A collection of data sets with the same base name, such as PAYROLL, that are kept in chronological order. Each data set is called a generation data set.

global spares. See spare devices.

L

large volume. A 3390-9 CKD volume defined with 32760 cylinders.

link address. An address assigned during initialization that identifies a channel or control unit so that the channel or control unit can send and receive frames, and perform I/O operations. See logical paths.

LLAPI. An ECAM device driver available to vendors which provides the ability to query an SVA subsystem and its devices as well as the ability to manipulate functional tracks.

logical array. A grouping of devices into an array. The grouping of devices does not depend on their physical location.

logical partition. The subset of a processor unit that is allocated to support the operation of a systems control program.

logical paths. The relationship between a channel and a control unit that designates the physical path to be used for device-level communication between the channel and the control unit. This relationship is defined within the channel and control unit by a link address assigned to the control unit and a link address assigned to the channel.

M

MAT partition. The SVA partition consisting of drive modules that are not yet available for storing user data. Drive modules are automatically members of the MAT partition when they are first physically inserted in the SVA or when they have been drained of data.

Media Acceptance Test partition. See MAT partition.

N

net capacity load (NCL). This number is two KB times the number of physical sectors actually used to store user data, not including redundancy data. NCL is a percentage of the total number of sectors that are storing user data and is based on physical capacity used.

nonquiesced snap. A snap taken when the system is in full read-write access mode.

nonvolatile storage (NVS). The redundant solid state memory in the Disk Array Controller that remains active when ac power is removed. NVS protects any data that has not been written to the disk arrays.

P

Parallel Access Volume. A combination of a real device (Base) and one or more pseudo-devices (Aliases) that together support multiple concurrent I/Os to enhance performance.

parallel channel. A channel that uses bus-and-tag cables to transmit data between the host and the Disk Array Controller.

partition. The logical separation of devices, arrays, or groups of arrays to allow different modes of operation. The SVA supports a MAT partition, a Test partition, a Production partition, a Spares partition, and an Unavailable partition.

Note: The Test partition is not available in the V2X, V960, or 9500 SVA.

PAV. See Parallel Access Volume.

physical capacity. The physical space contained in (1) a single drive module, (2) a partition, or (3) an SVA subsystem.

physical device. See drive module.

privileged ECAM device. Privileged ECAM devices are the only devices that SVAA can use to send messages to the subsystem to request a change in the SVA's state. Such messages include those that alter the subsystem configuration or start a drain.

At least one privileged ECAM device must be defined in each SVA; all functional volumes in an SVA subsystem can be defined as privileged ECAM devices.

Production partition. The SVA partition consisting of drive modules assigned to production arrays for storing user data.

PROFSIBA macro. The profile executed when the SVAA SIBADMIN program is started.

PROFSIBS macro. The profile executed when the SVAA Subsystem Reporting Program is started.

Q

quiesce. To end a process by allowing operations to complete normally.

quiesced snap. A snap taken while the system is quiesced; all buffered transactions are flushed to disk storage.

R

read hit. The situation in which data requested by the read operation is located in cache.

read miss. The situation in which data requested by the read operation is not located in cache.

reconstruction. See device reconstruction

redundancy group. A logical grouping of devices that are protected from data loss due to a device failure by the use of redundancy (parity) data that is stored across the devices. Arrays in the SVA are redundancy groups that protect data against two simultaneous device failures. See also: dual-redundancy disk array.

Reporter. The SVAA subsystem reporting program—the SVAA component that collects subsystem performance data and produces reports based on that data, as well as on space utilization.

S

SCSI channel. See SCSI I/O interface.

SCSI domain. An SVA addressing scheme, prefixed to SCSI target and LUN addresses, that extends the number of addressable volumes from SCSI-attached host systems.

serial channel. A channel that uses fiber-optic (ESCON) cables to transmit data between the host and the Disk Array Controller. See also: ESCON channel.

Service Information Message (SIM). A message generated by the host processor upon receipt of sense information from the SVA that contains notification of a need for repair or customer action, or status information.

Shared Virtual Array (SVA). StorageTek's online, random access disk array storage subsystem composed of a Disk Array Controller and 16 to 64 disk drive modules.

Shared Virtual Array Administrator (SVAA). StorageTek's host software product that enables implementation of the extended storage management facilities of the SVA, and offers additional functions including SnapShot, DDSR, and reporting capabilities.

SIBADMIN module. The module used to invoke SVAA in command mode.

SIBBATCH module. The SVAA module for batch configuration and reporting.

SIBDSR module. The SVAA module for deleted data space release (DDSR).

SIBIOATX exit. The Subsystem I/O Authorization user exit that verifies a user's authority to send a control message to a subsystem.

SIBSNDAX exit. The Command Authorization user exit that verifies a user's authority to send a command to a subtask in the SVAA address space.

SIBSRP module. The SVAA module for the subsystem reporting program.

SIBSRPSW exit. The Subsystem File Switch user exit that is invoked by the SVAA SRP data collection task before closing or switching a logging file.

SIBVOP module. The SVAA operator console command program.

SIM alert. An operator console message that alerts the operator that an action requiring attention has occurred.

slot. The physical location of an SVA subsystem drive module.

snap. (noun) A duplication of a source volume or data set with SnapShot (see SnapShot). A snap is also the result of a successful SnapShot operation (not the use of a data mover). Synonymous with SnapShot. Contrast with *data mover copy*.

snap. (verb) To duplicate a functional volume or data set with SnapShot.

SnapShot. StorageTek's high-speed data-duplication facility, available only with the SVA and packaged with SVAA. SnapShot achieves great time-savings in duplicating volumes or data sets because it only creates a second set of pointers to the data. No additional physical disk space is used in the process.

source. The minidisk or volume from which data is snapped.

spare devices. SVA drive modules that are physically installed but not logically associated with an array. Spare devices are used by the SVA to form new arrays or to automatically reconstruct and logically replace failed devices.

spares. See spare devices.

Spares partition. The SVA partition consisting of all of the spare devices in the subsystem. See spare devices.

SSID. See subsystem identifier (SSID)

Status Bridge. A pair of devices used by Power PPRC to transmit acknowledgements that the data was received at the other end. These devices are not used to store customer data.

storage cluster. A power and service region that processes channel commands and controls the data storage devices. The SVA contains two storage clusters, each of which contains interfaces for up to 16 channels.

Storage Management Subsystem (SMS). An IBM approach to storage management in which a host system determines data placement and an automatic data manager handles data backup, movement, space, and security.

subsystem free space. Storage space in the disk arrays that does not contain user data.

subsystem identifier (SSID). The identifier for a 3990 controller emulated within the SVA. From one to sixteen SSIDs (logical 3990s) can be defined in each subsystem. Within an installation, each logical 3990 is defined by a unique four-digit (hexadecimal) SSID.

subsystem reporting program (SRP). The SVAA component that collects subsystem performance data

and produces reports based on that data, as well as on space utilization. See also: Reporter.

SVAA profile facility. When invoked, this facility allows the user to specify commands for an SVAA session.

T

target. The minidisk or volume to which data is snapped.

Test partition. The SVA partition consisting of drive modules assigned to a test array and containing test data. The Test partition allows user-controlled, host-driven, testing of arrays, as though they were production arrays.

Note: The Test partition is not available in the V2X, V960, or 9500 SVA.

tray. The physical packaging of eight drive modules within the disk array area of the SVA.

U

Unavailable partition. The SVA partition consisting of drive modules that are not available for use in an array. Drive modules that are not installed or have failed are in this partition.

unit. See Disk Array Unit.

V

virtual cylinder. An operating system unit of measure available to allow a system administrator to view and manage the total amount of functional cylinders available in an SVA subsystem.

virtual device identifier (VDID). Another term for FDID. See functional device identifier (FDID).

virtual operator panel (VOP). An SVAA facility that allows operator interaction with, and control of, the SVA via a host operator console in lieu of the local operator panel (LOP), the DOP, or remote operator panel.

volatile memory. See cache volatile memory.

volume. See functional volume.

volume preferencing (VP). A facility of SnapShot that filters the choice of an SMS target volume for newly allocated data sets.

volume serial number. A six-character alphanumeric name that identifies a disk volume to the host operating system.

W

write hit. The situation in which data to be updated by a write operation is located in cache.

write miss. The situation in which data to be updated by a write operation is not located in cache.

Abbreviations and Acronyms

API	application programming interface
BFDID	base functional device ID
CCHH	cylinder-head address (CC is the two-byte cylinder number, HH is the two-byte head number
CCW	channel command word
CFW	cache fast write
CKD	count-key-data
CLI	command line interface
COD	Capacity on Demand
CSA	common service area
CSI	consolidated software inventory
DADSM	direct access device space management
DASD	direct access storage device
DAU	Disk Array Unit
DDSR	Deleted Data Space Release
DFP	Data Facility Product
DFSMS	Data Facility Storage Management Subsystem
DFW	DASD fast write
DLIB	distribution library
DOP	Detached Operator Panel
DSF	Data Support Facilities
DTL	domain-target-LUN
ECAM	Extended Control and Monitoring
ECSA	extended common service area
ESA	Enterprise Systems Architecture
ESCON	Enterprise Systems CONnection
ESDI	enhanced small device interface
FDID	functional device identifier
FICON	FIbre CONnection
FMID	function modification identifier
FRU	field-replaceable unit
FSC	fault symptom code, or free space collection
FTD	functional track directory
FTR	functional track recovery
GB	gigabyte
GDG	generation data group
GRS	Global Resource Serialization
GTF	generalized trace facility
HCD	hardware configuration definition
HSI	Host Subsystem Interface
ICKDSF	ICK Data Support Facilities
IDCAMS	IDC access method services
IDID	interface device identifier
IML	initial microprogram/microcode load
IOCP	I/O configuration program
IPL	initial program load
ISPF	Interactive System Productivity Facility
I/O	input/output
LBA	logical block address
LOP	Local Operator Panel
LUN	logical unit number
MAT	Media Acceptance Test
MB	megabyte
MIH	missing interrupt handler
MVS	Multiple Virtual Storage
NCL	net capacity load
NVS	nonvolatile storage
PAV	Parallel Access Volume
PCAP	physical capacity
PPRC	peer-to-peer remote copy
PTF	program temporary fix
RACF	Resource Access Control Facility
RAID	redundant array of inexpensive disks
REXX	Restructured Extended Executor
RFA	record format assist
RIM	resource initialization module
RMF	resource measurement facility
SAF	Security Access Facility
SCP	system control program
SCSI	small computer system interface
SFC	Server Framework Component
SIM	service information message
SMF	system management facility
SMP/E	System Modification Program Extended
SMS	Storage Management Subsystem
SRP	Subsystem Reporting Program

Abbreviations and Acronyms

SSID	subsystem identifier
SVA	Shared Virtual Array
SVAA	Shared Virtual Array Administrator
TSO	Time Sharing Option
UCB	unit control block
VCU	virtual control unit
VDID	virtual device identifier
volser	volume serial number
VOP	virtual operator panel
VP	volume preferencing
VSAM	Virtual Storage Access Method
VTOC	volume table of contents
VVDS	VSAM volume data set
VVR	VSAM volume record

Index

Special Characters

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Reader's Comment Form

Product Name: Shared Virtual Array Administrator for OS/390 **Software Level:** Version 3.1
Manual Name: Reporting **Document Number:** 311290608

Please check or fill in the items, adding explanations or comments in the spaces provided.

Which of the following terms best describes your job?

_ Field Engineer	_ Manager	_ Programmer	_ Systems Analyst
_ Engineer	_ Mathematician	_ Sales Representative	_ Systems Engineer
_ Instructor	_ Operator	_ Student/Trainee	_ Other (explain below)

How did you use this publication?

_ Introductory text _ Reference manual _ Student/Trainee _ Instructor text
_ Other (explain) _____

Did you find the material easy to read and understand?	_ Yes	_ No	(explain below)
Did you find the material organized for convenient use?	_ Yes	_ No	(explain below)

Specific criticisms (explain below):

Clarifications on pages _____

Additions on pages _____

Deletions of pages _____

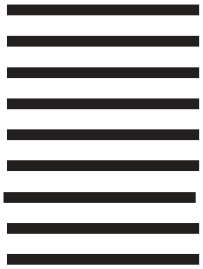
Errors on pages _____

Explanations and other comments:

Note: *Staples can cause problems with automated-mail sorting equipment. Please use pressure-sensitive tape to seal this form. If you would like a reply, please supply your name and address on the reverse side of this form. Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A.*

TO COMPLY WITH POSTAL REGULATIONS, FOLD EXACTLY ON DOTTED LINES AND TAPE (DO NOT STAPLE)

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES



BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 2 LOUISVILLE, CO U.S.A.

POSTAGE WILL BE PAID BY ADDRESSEE

STORAGE TECHNOLOGY CORPORATION
MANAGER, DISK STORAGE LEARNING PRODUCTS
ONE STORAGETEK DRIVE
LOUISVILLE, COLORADO 80028-2121
U.S.A.

FOLD HERE AND TAPE DO NOT STAPLE FOLD HERE AND TAPE

If you would like a reply, please print:

Your Name: _____

Company Name: _____ Department: _____

Street Address: _____

_____

Cut or Fold Along Line

Printed in U.S.A.

NEED MORE INFORMATION?

www.storagetek.com

ABOUT STORAGETEK

Storage Technology Corporation (NYSE: STK) is a $2 billion global company that enables businesses, through its information lifecycle management strategy, to align the cost of storage with the value of information. The company's innovative storage solutions manage the complexity and growth of information, lower costs, improve efficiency and protect investments. For more information, visit www.storagetek.com, or call 1.800.275.4785 or 01.303.673.2800.

WORLD HEADQUARTERS

Storage Technology Corporation
One StorageTek Drive
Louisville, Colorado 80028 USA
1.800.678.4430 or 01.303.673.4430

StorageTek equipment is manufactured from new parts, or new and used parts. In some cases, StorageTek equipment may not be new and may have been previously installed. Regardless, StorageTek's standard warranty terms apply, unless the equipment is specifically identified by StorageTek as "used" or "refurbished."

Replacement parts provided under warranty or any service offering may be either new or equivalent-to-new, at StorageTek's option. Specifications/features may change without notice.