

Unicenter[®] CA-Explore[®] Performance Management for CICS

History Reporting Guide

Release 7.0



Computer Associates[®]

K01102-1E

This documentation and related computer software program (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by Computer Associates International, Inc. ("CA") at any time.

This documentation may not be copied, transferred, reproduced, disclosed or duplicated, in whole or in part, without the prior written consent of CA. This documentation is proprietary information of CA and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, licensed users may print a reasonable number of copies of this documentation for their own internal use, provided that all CA copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the confidentiality provisions of the license for the software are permitted to have access to such copies.

This right to print copies is limited to the period during which the license for the product remains in full force and effect. Should the license terminate for any reason, it shall be the user's responsibility to return to CA the reproduced copies or to certify to CA that same have been destroyed.

To the extent permitted by applicable law, CA provides this documentation "as is" without warranty of any kind, including without limitation, any implied warranties of merchantability, fitness for a particular purpose or noninfringement. In no event will CA be liable to the end user or any third party for any loss or damage, direct or indirect, from the use of this documentation, including without limitation, lost profits, business interruption, goodwill, or lost data, even if CA is expressly advised of such loss or damage.

The use of any product referenced in this documentation and this documentation is governed by the end user's applicable license agreement.

The manufacturer of this documentation is Computer Associates International, Inc.

Provided with "Restricted Rights" as set forth in 48 C.F.R. Section 12.212, 48 C.F.R. Sections 52.227-19(c)(1) and (2) or DFARS Section 252.227-7013(c)(1)(ii) or applicable successor provisions.

© 2003 Computer Associates International, Inc.

All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.



Contents

Chapter 1: Using History Reporting

Welcome to Unicenter CA-Explore for CICS History Reporting	1-2
Memory and Partition Requirements	1-2
Concepts and Terminology	1-3
Command Syntax	1-5

Chapter 2: Planning Your Report

Step 1: Decide Whether to Use Short-Term or Long-Term Data	2-2
Flashback File (Short-Term Data)	2-2
Archive File or Log File (Long-Term Data)	2-3
Step 2: Decide Whether to Use a Canned Report	2-3
Step 3: Decide What Format to Use	2-4
Tabular Reports	2-5
Plot Reports	2-6
Graph Reports	2-7
Flashback Reports	2-8
Step 4: Decide What Information to Include	2-9
Variables	2-9
Variable Options	2-9
ID Options and Identifiers	2-10
Step 5: Decide How to Further Tailor Information	2-10

Chapter 3: Writing Your Report

Step 1: Create a Job Stream	3-2
Report Input from Flashback Files	3-2
Report Input from Archive Files	3-3
Report Input from Tape Log Files	3-4
Report Input from Disk Log Files	3-5
Specifying Report Output Preferences	3-5
Step 2: Write History Reporting Commands	3-6
Global Commands	3-7
Report Commands	3-7
Writing One Report	3-9
Writing Multiple Reports	3-10
Tailoring Reports Using Secondary Commands	3-10
Step 3: Incorporate Report Commands into the Job Stream	3-12

Chapter 4: Reading Reports

Reading Tabular Reports	4-1
Simple Tabular Reports	4-2
Complex Tabular Reports	4-3
TAB2 Reports	4-4
Reading Plot Reports	4-5
PLOT Command Horizontal Reports	4-5
PLOT2 Command Horizontal Reports	4-7
VLOT Command Vertical Reports	4-8
VLOT2 Command Vertical Reports	4-9
MLOT Command Vertical Reports	4-11
HLOT Reports	4-14
Reading Graph Reports	4-16
GRAF2 Reports	4-17
Reading Flashback Reports	4-18
Flashback List Reports	4-18
Flashback Detail Reports	4-19

Chapter 5: Canned Reports

Generating Canned Reports	5-1
Command Syntax and Operand	5-1
Command Expansions	5-2
Tailoring Canned Reports	5-3
Creating Your Own Canned Reports	5-5
Method 1: Add Reports to the Supplied Member	5-5
Method 2: Create an RDT Containing Your New Canned Reports	5-6
Canned Report Descriptions	5-7
CICSSTAT Report	5-11
CICS.DCT Report	5-12
CICS.DLIDBD Report	5-13
CICS.DLIPGM6 Report	5-14
CICS.DLIPGM7 Report	5-15
CICS.DLIPSB6 Report	5-16
CICS.DLIPSB7 Report	5-18
CICS.FILESTAT Report	5-20
CICS.FILEWAIT Report	5-22
CICS.INTERVAL.STAT Report	5-23
CICS.INTERVAL.STORAGE.STAT Report	5-24
CICS.MAXTIME Report	5-25
CICS.OPERSTAT Report	5-27
CICS.PERFORMANCE.RECORD.ANALYSIS.STAT Report	5-29
CICS.PROGUSE Report	5-33
CICS.REVIEW.STAT Report	5-34
CICS.STORAGE Report	5-36
CICS.SYSTEM.THRESHOLD.STAT Report	5-38
CICS.TEMPSTOR Report	5-39
CICS.TERMIO Report	5-41
CICS.TERMRESP Report	5-42
CICS.TERMRESP.RANGE Report	5-44
CICS.TERMSTAT Report	5-46
CICS.TERMTRAN Report	5-48
CICS.TIMESTAT Report	5-50
CICS.TRANFILE Report	5-52
CICS.TRANFIO Report	5-54
CICS.TRANGETM Report	5-56
CICS.TRANLIFE.RANGE Report	5-58
CICS.TRANPROG Report	5-60
CICS.TRANRESP Report	5-62
CICS.TRANSTAT Report	5-64

CICS.TRANSTOR Report	5-66
CICS.TRANTERM Report	5-67
CICS.TRANTIO Report	5-69
CICS.TRANUSE Report	5-71
CICS.TRANWAIT Report	5-72
VSE.CICS.STAT Report	5-74

Chapter 6: Command Descriptions and Syntax

Understanding Syntax Rules	6-2
Applying Syntax Conventions	6-3
Using Global Commands	6-4
Using Report Commands	6-5
Command Descriptions	6-7
CANRPT Command	6-7
CARDLEN Command	6-8
COMMENT Command	6-8
COST Command	6-9
COUNT Command	6-12
DEFINE Command	6-13
EACH Command	6-19
END Command	6-22
EXCLUDE Command	6-23
EXITRTN Command	6-24
FLASHBACK DETAIL Command	6-24
FLASHBACK LIST Command	6-25
FROM and TO Commands	6-27
GLOBAL Command	6-31
GOTO Command	6-32
GRAF Command	6-33
GRAF2 Command	6-35
GROUP Command	6-38
HEADER Command	6-39
HPLOT Command	6-41
IF Command	6-42
INCLUDE Command	6-43
INPUT Command	6-44
INVOICE Command	6-47
LINECOUNT Command	6-50
MPLOT Command	6-50
OPTION Command	6-52

OUTPUT Command	6-59
PERIOD Command	6-61
PLOT Command	6-63
PLOT2 Command	6-64
PRTEXT Command	6-65
RANGE Command	6-66
RUN Command	6-68
SELECT Command	6-68
SHIFT Command	6-70
TABulate Command	6-71
TAB2 Command	6-74
TITLE1, TITLE2, and TITLE3 Commands	6-75
VPLOT Command	6-78
VPLOT2 Command	6-79

Chapter 7: Variables

Structure of Variable Names	7-2
Variable Descriptions	7-3
Interval Record Type	7-3
PERformance Record Type	7-3
REVIEW Record Type	7-4
Summary Record Type	7-4
System Record Type	7-4
Units of Measure	7-4
INTerval Record Type for CICS	7-5
DATASET Variable Class	7-5
DSA Variable Class (Variables for CICS 2.3 and TS 1.1)	7-6
More About CICS TS 1.1 Dynamic Storage Areas	7-9
GETVIS Variable Class	7-10
REGION Variable Class	7-12
TRANsaction Variable Class	7-13
PERformance Record Type	7-15
COMMAND Variable Class	7-15
DATASET Variable Class	7-17
DLI Variable Class	7-21
EXIT (Global User Exits) Variable Class	7-25
IRC (Inter-Region Communication) Variable Class	7-26
ISC (Intersystem Communication) Variable Class	7-28
JOURNAL Variable Class	7-29
PROGram Variable Class	7-31

RESOURCE Variable Class	7-33
SECURITY Variable Class	7-35
SQL Variable Class	7-37
STORAGE Variable Class	7-38
TABLE Variable Class	7-39
TEMPSTOR Variable Class	7-40
TERMinal Variable Class	7-42
THRESHOLD Variable Class	7-43
TOTAL (TOTAL Database) Variable Class	7-45
TRANsaction Variable Class	7-47
TRANSIENT Variable Class	7-55
WAIT Variable Class	7-56
REVIEW Record Type	7-57
DATASET Variable Class	7-57
DSA Variable Class	7-57
TERMinal Variable Class	7-58
THRESHOLD Variable Class	7-58
TRANsaction Variable Class	7-59
SUMmary Record Type	7-60
TERMinal Variable Class	7-60
TRANsaction Variable Class	7-61
SYStem Record Type	7-64
THRESHOLD Variable Class	7-64

Chapter 8: Variable Options

Descriptions of Variable Options	8-2
COUNT Option	8-2
FORMAT Option	8-2
LOGICAL Option	8-4
MAX and MIN Options	8-5
MAXDATE and MAXTIME Options	8-6
PerCenT Option	8-6
RATE Option	8-6
SELZERO Option	8-8
TOTAL Option	8-8
XTIME, XTIMEM, and NXTIME Options	8-9
XTND and NXTND Options	8-9

Chapter 9: ID Options

Descriptions of ID Options	9-2
Using ID Options With Commands	9-7
TAB Report Using an ID Option and Identifiers	9-9
TAB and TAB2 Report Using ID Options	9-10
Graph Report Using an ID Option Without Identifiers	9-11
Graph Report Using an ID Option and Identifiers	9-12
Flashback Report Using an ID Option and Identifiers	9-13
Using ID Options with Variables	9-14
Using ID Options With Commands Versus With Variables	9-15
Using ID Options With Both Commands and Variables	9-17
Formatting Identifiers in Output	9-20
Using the FORMAT Option	9-20
Using the DEFINE Command	9-22
Using the DEFINE Command Versus the FORMAT Option	9-23

Chapter 10: Tabular Reports

TABulate Command	10-2
Organizing Reports by Time and Date or Resources	10-5
Tabulating by Date and Time	10-5
Tabulating by Time	10-6
Tabulating by Date	10-7
Tabulating by Day	10-7
Tabulating by Resources	10-8
Tabulating by Both Time or Date and Resources	10-9
Specifying Variables With or Without ID Options	10-10
Using Both CICS and VSE Variables	10-11
Using ID Options with Variables	10-12
Changing Column Headings	10-13
Using Variable Options	10-14
Displaying Percentage of Change Over Time	10-14
Changing a Count to a Rate	10-16
Specifying a Time Period for a Variable	10-17
Shifting a Column to the Right	10-18

Limiting the Resources Included in a Report	10-19
Using the FOR Operand to Limit Reports Organized by Identifier	10-20
Using the FOR Operand to Limit Reports Organized by Time and Date	10-21
Using the TOP or BOTTOM Operand to Limit TAB Reports	10-22
Sorting Reports	10-24
Sorting Reports by Different Keys	10-25
Sorting Reports in Ascending Order	10-26
Sorting Reports by Identifier	10-28
Using Other Commands with the TAB Command	10-29
Tabulating by Ranges	10-29
Tabulating by Groups	10-32
Tabulating by Both Group and Range	10-33
Creating Charge-Back Reports	10-34
TAB2 Reports	10-36
TAB2 Command	10-36
Using the FOR Operand to Limit TAB2 Reports	10-37
Using the TOP or BOTTOM Operand to Limit TAB2 Reports	10-39

Chapter 11: Plot Reports

Horizontal Reports (PLOT and PLOT2)	11-2
PLOT Command	11-2
PLOT2 Command	11-10
Vertical Reports (VPLOT, VPLOT2, and MPLOT)	11-13
VPLOT Command	11-13
VPLOT2 Command	11-19
Distribution Plot Reports (MPLOT)	11-22
MPLOT Command	11-22
Horizontal Reports (HPLOT)	11-29
HPLOT Command	11-29
Using VPLOT with HPLOT	11-30
Using MPLOT with HPLOT	11-32

Chapter 12: Graph Reports

GRAF Reports (GRAF and GRAF2)	12-2
GRAF Command	12-2
Creating a Simple GRAF Report	12-4
Graphing by an Alternate Resource	12-5
Limiting Reports Using ID Options	12-6
Limiting Reports Using the FOR Operand	12-7
Changing the Scale of a Report	12-8
Sorting Reports	12-9
Creating Reports on Specified Shifts	12-10
Creating Graphs with Multiple Time Periods	12-11
Limiting Input to a Specified Range	12-12
GRAF2 Reports	12-13
GRAF2 Command	12-13
Limiting GRAF2 Reports Using the FOR Operand	12-16
Tailoring GRAF2 Reports Using the SCALE Operand	12-17
Graphing by an Alternate Resource	12-18
Comparing Performance Using Multiple-Variable Criteria	12-19
Changing the Plot Characters	12-20

Chapter 13: Flashback Reports

FLASHBACK LIST Reports	13-2
FLASHBACK LIST Command	13-2
Default FLASHBACK LIST Report	13-3
Using ID Options to Limit FLASHBACK LIST Reports	13-4
Changing Columns by Specifying Variables	13-6
Using the Date and Time Operands	13-7
Using the RANGE Command to Limit Reports	13-8
FLASHBACK DETAIL Reports	13-9
FLASHBACK DETAIL Command	13-9
Default FLASHBACK DETAIL Report	13-9

Appendix A: User Exits

User Exit Execution	A-1
Source Code	A-2

Index

Using History Reporting

Unicenter® CA-Explore® Performance Management for CICS (Unicenter CA-Explore for CICS) provides batch-reporting facilities that let you produce easy-to-run, canned and customized reports.

This guide explains how to use the history reporting commands and variables to create reports. It assumes that the person writing reports has a functional knowledge of VSE, can edit a file, and can access the flashback and log files, which are described in the chapter titled “Planning Your Reports.”

Note 1: This guide discusses the uses of Unicenter CA-Explore for CICS with CICS Version 2.3 and CICS Transaction Server Version 1.1 (TS 1.1). Unless a distinction is noted, the information in this guide applies to both versions.

Note 2: Any reference in this guide to SQL should be translated to DB2. The term *SQL* has been retained intentionally for backward compatibility.

Note 3: This manual describes version 6.7 of the Unicenter CA-Explore for CICS report writer, which is supplied with version 7.0 of Unicenter CA-Explore for CICS.

Welcome to Unicenter CA-Explore for CICS History Reporting

The Unicenter CA-Explore for CICS History Reporting subsystem, or *report writer*, lets you produce graphic and tabular reports from the performance data collected by Unicenter CA-Explore for CICS. With these reports, you can study resource use on a long-term basis, and detect and analyze unusual events.

The Unicenter CA-Explore for CICS report writer lets you report on a variety of CICS activities, such as transaction use and maximum terminal response time. You control the time that each detail line in a report represents and the time frame that a report covers. The report writer lets you access the information you want, at the level of detail you indicate. With just one pass through the historical data, you can create up to 50 reports, each covering a different time.

Memory and Partition Requirements

The amount of virtual storage needed by the report writer depends upon the types of reports to be produced, the number of reports requested in each run, and the amount of input data used. To reduce the amount of virtual storage required, limit the number of reports in each run or narrow the period covered by each report.

It is recommended that you run the report writer in a partition of at least 1 megabyte.

Concepts and Terminology

Becoming familiar with these will enable you to use the report writer more efficiently. The following table describes report writer commands and terms, which are described in detail in the chapter titled “Planning Your Reports:”

Command or Term	Purpose	Examples
Report writer commands	Used with JCL to produce reports. Report writer commands include global commands and report commands.	INPUT, RUN, CANRPT, PLOT, EACH, TITLE1
Global commands	Report writer commands that affect all reports in a run.	COUNT, INPUT, RUN
Report commands	Report writer commands that affect only a single report. Report commands include primary report commands and secondary report commands.	CANRPT, PLOT, EACH, TITLE1
Primary report commands	Report commands that define the type of report to be generated.	CANRPT, PLOT, TAB2
Secondary report commands	Report commands that alter or enhance a report.	EACH, FROM, TITLE1
Operands	Keywords specified with commands or variable options that affect the content or structure of a report. Different commands and variable options have different operands.	The syntax of the CANRPT command is <code>CANRPT reportname</code> . Its operand is <i>reportname</i> .
Variables	Structured keywords that specify what a report is to be about.	TRAN USE, TERM RESP

Command or Term	Purpose	Examples
Variable options	Keywords that change either the type of data a variable represents, or the format of that data.	AVeraGe, FORMAT, RATE
ID options	Keywords that change the scope of the data that a variable represents, or organize the data in a tabular or flashback report by a type of resource.	TRANID, TERMID, USERID
Identifiers	Either the name of a particular resource, or a generic specification of a group of resources that a report or variable is to be limited to. Identifiers are specified following ID options.	MYTRAN, -OPER1, TERM1+, CICS*

Command Syntax

History reporting commands are specified using conventions, or syntax.

Sample Syntax

This example of the PLOT command illustrates the command syntax used in this guide.

```
PLOT variable [options] [ALONE] [SCALE({n},{mX})] [CHAR1(x)]
                                         [CHAR2(x)]
```

The following table describes command syntax elements:

This Syntax	Indicates	Examples
Lowercase italicized letters	Variables, or items you are to replace with specific information	PLOT <i>variable</i> where you replace <i>variable</i> with a word or value.
All capital letters	Literals, or items you must enter as shown.	PLOT <i>variable</i> where you must enter PLOT as it is shown.
Square brackets	Optional operands	[<i>options</i>] [ALONE] where you can choose from several options to replace <i>options</i> , or enter the ALONE option as shown.
Braces	Parameters from which you must choose only one	{ <i>n</i> },{ <i>mX</i> }
Ellipses	Variables that are part of a series, but are not shown	<i>n1</i> , ..., <i>n10</i>

Ellipses in Reports

Vertical ellipses are used within reports to indicate that only part of a report is shown, as illustrated here:

JOB NAME	JOB COUNT	STEP COUNT	TOTAL DURATION	AVERAGE DURATION	TCPU TIME	CPU TIME	DISK SIO	SERVICE TIME	TAPE IO	LINES SPOOLED	PAGES SPOOLED	MAX GETVIS	MAX %GETVIS
COMPLK	3	3	00:00:47	00:00:15	1.667	0.417	930	0.040	0	577	25	348K	7.6
CREADS	2	2	00:01:37	00:00:48	0.860	0.430	232	0.046	0	4,816	122	72K	1.7
CATALR	2	2	00:00:14	00:00:07	0.249	0.124	256	0.043	0	76	4	48K	1.2
AXPHJ6F	1	5	00:00:39	00:00:07	0.616	0.123	732	0.040	0	108	9	312K	1.5
AXPHJ6B	1	5	00:00:25	00:00:05	0.548	0.110	710	0.030	0	107	9	296K	1.5
GSJOBCTL	1	3	00:00:25	00:00:08	0.539	0.179	655	0.029	0	88	4	308K	8.1
CICSMROA	1	2	01:32:51	00:46:25	7.741	0.860	763	0.036	0	55,875	1,692	472K	11.9
ERRM	1	1	00:01:39	00:01:39	7.875	7.875	685	0.055	0	6,224	157	68K	1.8
EVSEBKUP	1	1	00:00:28	00:00:28	1.242	1.242	656	0.032	0	58	2	204K	5.0
FAQSDSPL	1	1	00:00:08	00:00:08	0.560	0.560	276	0.024	0	65	3	296K	7.8
LIBR	1	1	00:00:08	00:00:08	0.141	0.141	86	0.057	0	276	10	16K	0.4
PAUSE	1	0	00:00:00	00:00:00	0.000	0.000	0	0.000	0	29	1	0	0.0
NICK2	1	0	00:00:00	00:00:00	0.000	0.000	0	0.000	0	14	2	0	0.0
SETSDL	1	0	00:00:00	00:00:00	0.000	0.000	0	0.000	0	7	1	0	0.0
.
.
.
SUMMARY	22	61	09:37:26	00:09:27	75.41	0.711	23,654	0.036	45	68,575	2,061	2664K	90.2

Planning Your Report

This chapter describes the steps you take in planning your report—what it should include, what it should look like, and the range of time on which it should report.

Before you begin to write a report, you must make the following decisions:

Step	Action
1	Decide whether you want to use short-term or long-term data. This determines which file you access and the job stream you use to run the report. For short-term data, use the flashback file. For long-term data, use either the archive file or the log file (a backup of the archive file).
2	Decide whether it would be easier to use a report writer <i>canned</i> (predefined) reports; one of them might provide exactly the information you want. If so, use a canned report rather than write your own history reporting commands.
3	Decide what format you want for the report: tabular, plot, graph, or flashback. Examples of each of these are shown later in this chapter. The format determines the primary report command that you use.
4	Decide what information you want in the report, which resources you want this information about, and how you want the information ordered and formatted. For example, you might want to create a tabular report showing all transactions, in alphabetical order, how many times each transaction was used, and the average transaction lifetime. To select this information, you use variables, variable options, ID options, and identifiers.
5	Decide how to further tailor your report using secondary commands. Such commands can limit the time a report covers, group the resources included in the report, or assign the report a descriptive title.

Step 1: Decide Whether to Use Short-Term or Long-Term Data

When you run Unicenter CA-Explore for CICS, you can specify that historical data be collected into either or both of the following VSAM disk files:

- **Flashback file (EXPCFBK)** – A wrap-around disk file used for short-term analysis, including batch reports, online reports, and online flashback analysis.
- **Archive file (EXPCARC)** – Backed up to tape or disk to create a log file for long-term analysis. Performance data can be appended to the end of the log file.

You can then use the data in the flashback file, the archive file, or the log file to write reports. Logging data to the flashback or archive file is optional, but of course, you cannot generate history reports if there is no historical data available.

Note: You can generate reports from data collected by Unicenter CA-Explore for CICS Versions 3, 4, 5, and 6.

Flashback File (Short-Term Data)

Because you can create reports directly from the flashback file, you can use it to quickly generate reports on recent data. The flashback file is a wrap-around file; it writes over old data as soon as its disk space is depleted. How quickly the flashback file begins to wrap depends on its size, the settings of your Unicenter CA-Explore for CICS configuration options, and your site's typical work load.

You can increase or decrease the allocation for your flashback file to meet your needs. To increase or decrease the size of your flashback file, use the `INSTALL` utility option. For more information, see the *Unicenter CA-Explore for CICS Utilities Guide*.

Archive File or Log File (Long-Term Data)

The performance information in the archive file can cover an extended period. This period is user-defined.

You can either create a report directly from the archive file, or you can first back up the archive file to tape or disk to create a log file and use the log file as input for your reports. (Normally, you back up the archive file to tape or disk when it becomes full, so you do not lose data.)

You can create several log files, or you can merge data from several archive files into one log file using the EXPUTIL program.

For information on using the EXPUTIL program to back up or merge archive files, see the Unicenter CA-Explore for CICS *Utilities Guide*.

Limiting the Time Required for Running Reports

Because the log file can contain a great deal of data, generating a report that uses it can take a long time. To limit output when testing reports, either use the flashback file, or specify the COUNT parameter (card) to limit the records read from your log file.

Step 2: Decide Whether to Use a Canned Report

The report writer includes a number of predefined, or *canned* reports, which are described in the chapter titled “Canned Reports.” Look through these descriptions. If you find a report similar to the one you want to produce, and if you are new to the report writer, use the canned report.

Canned reports are easy to produce because the commands are already written for you. The report writer shows you the commands and allows you to tailor them in certain ways. However, once you are more experienced with the report writer, you may want to write your own report commands to create reports that suit your needs more precisely.

Step 3: Decide What Format to Use

If you decide not to use a canned report, you must write the commands to produce your report. Whether you produce your reports from the flashback file or the archive file, you can generate reports in four formats: tabular, plot, graph, and flashback. Detailed information about creating reports in each of these formats is provided in dedicated chapters later in this guide. For information about reading each type of report, see the chapter titled “Reading Reports.”

The sections that follow provide a sample of each report format.

Tabular Reports

Tabular reports display information in a tabular format. Use this type of report to compare many variables at once, as shown in the following example.

FILE ID	DL/I USE	DL/I SERVTIME	DL/I SIOS	DL/I SIO TIME	DL/I READS	DL/I GU	DL/I GET NEXT	DL/I GHU	DL/I GHN	DL/I WRITES	DL/I REPLACES	DL/I INSERT	DL/I DELETES
DLZBPC00	385K	0.006	81769	0.026	385K	0	0	194K	51825	0	0	0	0
GHU	198K	0.004	31126	0.023	198K	0	0	198K	0	0	0	0	0
GHNP	145K	0.006	31122	0.029	145K	0	0	0	0	0	0	0	0
GHN	52252	0.012	23287	0.026	52256	0	0	0	52256	0	0	0	0
C02001	4134	0.003	481	0.025	3907	0	0	490	0	0	0	0	0
C00023	2473	0.015	1401	0.026	2148	0	0	1031	0	0	0	0	0
C02006	1080	0.007	302	0.025	814	0	0	636	0	266	105	3	158
C10001	953	0.009	346	0.024	552	0	0	302	0	349	349	0	0
REPL	619	0.001	12	0.020	0	0	0	0	0	628	628	0	0
C155001	594	0.003	115	0.017	594	0	0	405	0	0	0	0	0
C13004	446	0.006	91	0.028	363	0	0	203	0	0	0	0	0
C04007	394	0.023	374	0.024	389	0	0	80	309	0	0	0	0
C155010	390	0.014	271	0.020	390	0	0	199	0	0	0	0	0
A67201	340	0.005	64	0.025	338	0	0	120	74	0	0	0	0
C04005	302	0.010	118	0.025	228	0	0	134	0	64	64	0	0
C09007	234	0.016	148	0.026	111	0	0	71	0	52	26	0	26
C00000	234	0.001	6	0.021	156	0	0	78	0	0	0	0	0
C07001	181	0.002	12	0.029	114	0	0	37	0	50	50	0	0
C10007	80	0.005	16	0.026	40	0	0	31	0	0	0	0	0
ISRT	78	0.018	72	0.020	0	0	0	0	0	187	0	0	187
A67100	72	0.010	25	0.030	68	0	0	44	0	0	0	0	0
C155008	66	0.033	177	0.012	55	0	0	8	47	6	6	0	0
C155011	62	0.011	23	0.030	46	0	0	17	0	14	14	0	0
C155009	48	0.001	2	0.029	32	0	0	18	0	6	6	0	0
A91046	42	0.005	12	0.019	39	0	0	6	0	0	0	0	0
A91060	40	0.000	1	0.008	37	0	0	3	0	0	0	0	0
C04004	36	0.007	12	0.020	32	0	0	16	0	4	4	0	0
A85038	35	0.003	3	0.030	30	0	0	10	0	0	0	0	0
C15001	21	0.006	6	0.022	10	0	0	10	0	4	4	0	0
A67288	20	0.009	7	0.025	16	0	0	4	0	0	0	0	0
A85054	20	0.005	5	0.020	16	0	0	6	0	0	0	0	0
C13003	18	0.013	9	0.025	12	0	0	9	0	0	0	0	0
A91044	18	0.004	4	0.019	15	0	0	3	0	0	0	0	0
A91061	16	0.006	4	0.025	13	0	0	3	0	0	0	0	0
C155006	11	0.005	3	0.019	10	0	0	6	0	0	0	0	0
A85037	10	0.014	4	0.035	6	0	0	4	0	0	0	0	0
A85039	10	0.003	3	0.012	9	0	0	2	0	0	0	0	0
C13010	10	0.000	0	0.000	10	0	0	5	0	0	0	0	0
A91053	8	0.009	3	0.025	4	0	0	2	0	3	0	0	3
C13008	7	0.000	0	0.000	3	0	0	3	0	0	0	0	0
C13005	6	0.017	4	0.025	4	0	0	4	0	0	0	0	0
C07002	5	0.013	2	0.032	4	0	0	3	0	0	0	0	0
SUMMARY	794K	0.006	171K	0.026	791K	0	0	396K	105K	1633	1256	3	374

Flashback Reports

Flashback reports present a formatted listing of system events in the order they occurred. You can use flashback reports to discover the source of a recent problem, using data from either the flashback file or the log file. The FLASHBACK command produces a report similar to the following:

CICS ID	END TIME	TERM ID	OPER ID	TRAN ID	TRAN NUMBER	TRAN LIFE	TRAN CPU	TERM RESP	TERM IO	TRAN I/O	FILE COUNT	FILE USE	PROG COUNT	ABEND CODE	
A1	21.16.59	CNSL	N/A	CSDA	19	0.016	0.011	0.000	1	0.000	0	0	1		
A1	21.21.52	CNSL	N/A	CIRB	20	13.30	1.175	4.074	4	0.000	0	0	4		
B1	21.22.41	N/A	N/A	TCP	0	911.9	2.110	0.000	0	0.381	1	241	11		
A1	21.24.20	N/A	N/A	TCP	0	912.3	1.051	0.000	0	0.000	0	0	4		
B1	21.22.42	N/A	N/A	KCP	0	912.3	1.878	0.000	0	0.000	0	0	0		
B1	21.23.07	N/A	N/A	JJJ	0	937.1	1.942	0.000	0	0.000	0	0	0		
B1	21.23.07	N/A	N/A	CATD	51	6.529	0.326	0.000	0	0.000	0	0	5		
B1	21.23.08	N/A	N/A	CSNE	52	0.081	0.068	0.000	0	0.000	0	0	3		
B1	21.23.09	ZA00	N/A	CSGM	53	0.264	0.010	0.013	2	0.000	0	0	1		
B1	21.23.15	N/A	N/A	VGBR	55	0.332	0.023	0.000	0	0.000	0	0	2		
B1	21.23.21	N/A	N/A	CSNE	56	0.013	0.002	0.000	0	0.000	0	0	3		
B1	21.23.41	X250	N/A	VGBQ	57	19.33	0.151	0.389	5	0.000	0	0	7	AKCT	
B1	21.28.19	ZA00	N/A	VGBS	54	306.6	0.633	0.237	16	0.000	0	0	8		
B1	21.28.21	ZA00	N/A	CSAC	58	0.647	0.015	0.647	2	0.000	0	0	1		
B1	21.28.23	ZA00	N/A	END	59	0.486	0.232	0.000	1	0.000	0	0	1		
B1	21.28.24	N/A	N/A	CSNE	60	0.020	0.003	0.000	0	0.000	0	0	3		
B1	21.28.25	N/A	N/A	CATD	61	0.128	0.058	0.000	0	0.000	0	0	4		
B1	21.37.54	N/A	N/A	TCP	0	913.0	0.542	0.000	0	0.000	0	0	0		
A1	21.24.21	N/A	N/A	KCP	0	912.3	0.795	0.000	0	0.000	0	0	0		
A1	21.39.33	N/A	N/A	TCP	0	912.1	0.449	0.000	0	0.000	0	0	0		
B1	21.37.55	N/A	N/A	KCP	0	913.0	0.727	0.000	0	0.000	0	0	0		
B1	21.53.06	N/A	N/A	TCP	0	911.7	0.674	0.000	0	0.000	0	0	0		
A1	21.39.34	N/A	N/A	KCP	0	912.1	0.467	0.000	0	0.000	0	0	0		
A1	21.54.45	N/A	N/A	TCP	0	912.2	0.434	0.000	0	0.000	0	0	0		
B1	21.53.07	N/A	N/A	KCP	0	911.7	0.563	0.000	0	0.000	0	0	0		
B1	22.08.18	N/A	N/A	TCP	0	912.0	0.769	0.000	0	0.000	0	0	0		
A1	21.54.45	N/A	N/A	KCP	0	911.8	55.19	0.000	0	0.000	0	0	0		
A1	22.09.57	N/A	N/A	TCP	0	912.2	0.422	0.000	0	0.000	0	0	0		
B1	22.08.19	N/A	N/A	KCP	0	912.0	0.601	0.000	0	0.000	0	0	0		
B1	22.23.31	N/A	N/A	TCP	0	912.7	0.763	0.000	0	0.000	0	0	0		
A1	22.09.58	N/A	N/A	KCP	0	912.2	68.16	0.000	0	0.000	0	0	0		
A1	22.25.09	N/A	N/A	TCP	0	912.3	0.430	0.000	0	0.000	0	0	0		
B1	22.23.32	N/A	N/A	KCP	0	912.7	0.604	0.000	0	0.000	0	0	0		
B1	22.38.43	N/A	N/A	TCP	0	912.2	0.766	0.000	0	0.000	0	0	0		
A1	22.25.10	N/A	N/A	KCP	0	912.2	57.63	0.000	0	0.000	0	0	0		
A1	22.40.21	N/A	N/A	TCP	0	911.9	0.435	0.000	0	0.000	0	0	0		
-----					22.40.21	655	593.1	5.559	1.005	31	0.010	0	241	2	

Step 4: Decide What Information to Include

After you determine the format of your report, you must specify what information to include and which resources you want that information about.

- To specify what information you want, and in what format, you use variables and variable options. For example, you can create a report showing transaction use, and you can have this information presented as the number of occurrences or as percentages. Detailed information about variables and variable options are provided in dedicated chapters later in this guide.
- To specify which resources you want information about, you use ID options and identifiers. Detailed information about ID options and identifiers are provided in dedicated chapters later in this guide.

The following sections briefly describe how to use variables, variable options, ID options, and identifiers.

Variables

Use *variables* to identify the information to be included in a report, and in what order. If, for example, you want a report that plots the number of transactions executed on your system, you would use the following command, which uses the variable TRANSACTION USE:

```
PLOT TRANSACTION USE
```

For tabular and graph reports only, if you are running both Unicenter CA-Explore for CICS and Unicenter CA-Explore for VSE, you can use both CICS and VSE variables. VSE variables are described in the *Unicenter CA-Explore for VSE History Reporting Guide*.

Variable Options

Use *variable options* to change either the type of data a variable represents or the format of that data. For example, by default the variable TRANSACTION USE gives the number of transactions executed. You can use the variable option RATE with this variable to plot the rate at which transactions were executed instead, as follows:

```
PLOT TRANSACTION USE RATE
```

ID Options and Identifiers

Use *ID options* and *identifiers* to specify which resources are to be included in a report. You can specify ID options and identifiers with variables or with commands, as follows:

You can limit a particular variable to certain individual resources by specifying an ID option and one or more identifiers with the variable. For example, if you want to plot the number of times the transaction Cinq was executed, you can use the following command, which includes the ID option TRANID and the identifier Cinq:

```
PLOT TRANSACTION USE TRANID(Cinq)
```

You can limit an entire tabular, flashback, or graph report to resources of a given type by specifying an ID option with the report command. You can further limit the report to individual resources of that type by specifying identifiers with the ID option. For example, the following command generates a tabular report showing the number of times the transactions Cinq and Pinq were executed and the average lifetime of each:

```
TAB TRANID(Cinq,Pinq) TRANSACTION USE AND TRANSACTION LIFE
```

Step 5: Decide How to Further Tailor Information

In addition to restricting your report to certain variable values for specified resources, you can further tailor the report by using secondary report commands.

For example, you can use the commands TO and FROM to limit a report to only those transactions that were executed between 9 and 11 a.m. on June 1. You can also use the EACH command to define the period each detail line in a report is to represent, and you can use the TITLE1, TITLE2, and TITLE3 commands to specify a multi-line title for the report.

See the chapter titled “Commands” for a description of secondary report commands.

Writing Your Report

This chapter describes the general procedure for writing and submitting any history report or series of reports.

See the chapter titled “Reading Reports” for information about reading reports generated by the report writer.

The following table describes the general steps you must complete to produce a report:

Step	Action
1	Create a job stream. The job stream varies, depending on whether you take the report input from the flashback file or a log file. Sample JCL for each of these cases is provided in the detailed description of Step 1.
2	Specify the commands required to create your report.
3	Combine the commands with the job stream.
4	If you plan to use input from a tape log file, mount the tape containing the log file. The log file can be a disk file, i.e., the flashback file.
5	Run the job stream created in Step 3.

The topics that follow provide detailed instructions for completing Steps 1 through 3.

Step 1: Create a Job Stream

You can produce reports using either flashback files, archive files, or log files (backups of the archive file to tape or disk), as input to the report. JCL for each input-file type is furnished in the topics that follow. Use the version of JCL that is appropriate for your type.

You can customize your reports by specifying additional parameters in the EXEC statement. You can choose parameters that allow you suppress the introductory report banner, force all output to upper case, print only lines containing report data, set the date format, and specify whether to list report commands before printing the report. Refer to the topic, Specifying Report Parameters in the EXEC Statement, following the JCL descriptions.

Once you have defined the JCL for your report, write your history reporting commands. These commands are described later in Step 2: Write History Reporting Commands.

After writing your commands, incorporate these commands into the JCL, as described in Step 3.

Report Input from Flashback Files

Use the sample job stream below to write reports that use flashback file data as input:

```
* $$ JOB
* $$ LST CLASS=A
// JOB FBCKRPT      PERFORMANCE MONITOR REPORT WRITER
(1) // DLBL EXPCFBK, 'expcfbk.file', ,VSAM
(2) // LIBDEF *,SEARCH=expc.library
// EXEC EXPRPT,SIZE=EXPRPT
INPUT(EXPCFBK)
*****
***
(3) ** Your history reporting commands go here. **
***
*****
/*
/&
* $$ EOJ
```

Specify the following information (corresponding to the numbered lines in the preceding figure) to tailor your job stream:

1. You must include this DLBL statement. The standard DLBL filename for the Unicenter CA-Explore for the CICS flashback file is EXPCFBK. Replace *expcfbk.file* with the dataset name of the flashback file being used as input.
If you are running multiple regions, include a DLBL statement for each flashback file you want to use as input.
2. You must have a LIBDEF statement pointing to the Unicenter CA-Explore for CICS libraries (*expc.library*).
3. After the INPUT command, insert your history reporting commands.

Report Input from Archive Files

Use the sample job stream below to write reports that use archive file data.

```

* $$ JOB
* $$ LST CLASS=A
// JOB FBCKRPT      PERFORMANCE MONITOR REPORT WRITER
(1) // DLBL EXPCARC,'expcarc.file',,VSAM
(2) // LIBDEF *,SEARCH=expc.library
// EXEC EXPRPT,SIZE=EXPRPT
(3) INPUT (EXPCARC) (3)
(4) OPTION(MASTER=jobname) (4)
*****
***
(5) ** Your history reporting commands go here. ** (5)
***
*****
/*
/&
* $$ E0J

```

To tailor your job stream, specify the following information (corresponding to the numbered lines in the preceding figure):

1. You must include this DLBL statement. The standard DLBL filename for the Unicenter CA-Explore for CICS archive file is EXPCARC. Replace *expcarc.file* with the dataset name of the archive file being used as input.
2. You should have a LIBDEF statement pointing to the Unicenter CA-Explore for CICS libraries (*expc.library*).
3. This statement is required for generating reports from the archive file.
4. Include this OPTION statement when using archive file input to ensure that the report writer reads the most recent data logged. Replace *jobname* with the jobname of the master region.
5. After the INPUT command, insert your history reporting commands.

Report Input from Tape Log Files

Use the sample job stream below to write reports that use data from a backup of the archive file to tape.

```

* $$ JOB
* $$ LST
// JOB ARCRPT PERFORMANCE MONITOR REPORT WRITER
(1) // TLBL filename, 'expcarc.backup' (1)
(2) // LIBDEF *,SEARCH=expc.library (2)
// ASSGN SYS004,TAPE ASSIGNING INPUT FILE
// PAUSE MOUNT INPUT TAPE ON SYS004
// MTC REW,SYS004
// EXEC EXPRPT,SIZE=EXPRPT
(3) INPUT(filename) (3)
*****
***
** Your history reporting commands go here. ** (4)
***
*****
/*
/&
* $$ E0J

```

Specify the following information (corresponding to the numbered lines in the preceding figure) to tailor your job stream:

1. You must include this TLBL statement. Replace *filename* with the filename of your tape log file. Replace *expcarc.backup* with the dataset name of the file. If you are running multiple regions, include a TLBL statement for each tape log file you want to use as input.
2. You should have a LIBDEF statement pointing to the Unicenter CA-Explore for CICS libraries (*expc.library*).
3. Replace *filename* with the filename of your tape log file. If you do not specify an INPUT statement, the report writer defaults to TAPEIN, which causes the report writer to expect input from tape, and to process the file in the same manner as if you had specified the filename TAPEIN.
4. After the INPUT command, insert your history reporting commands.

Report Input from Disk Log Files

Use the sample job stream below to write reports that use data from a backup of the archive file to disk.

```

* $$ JOB
* $$ LST
// JOB ARCRIPT PERFORMANCE MONITOR REPORT WRITER
(1) // DLBL DISKxxx,' expcarc.backup'
(2) // LIBDEF *,SEARCH=expc.library
// ASSGN SYS004,DISK,VOL=SYSWK9,SHR
// EXEC EXPRPT,SIZE=EXPRPT
(3) INPUT(DISKxxx)
*****
***
(4) ** Your history reporting commands go here. **
***
*****
/*
/&
* $$ E0J

```

To tailor your job stream, specify the following information (corresponding to the numbered lines in the preceding figure):

1. Replace `DISKxxx` with the filename of your disk log file. The filename `DISKLOG` is reserved for Unicenter CA-Explore for CICS, and is valid only for Version 3.

Replace `expcarc.backup` with the dataset name of the file. If you are running multiple regions, include a `DLBL` statement for each disk log file you want to use as input.

2. You should have a `LIBDEF` statement pointing to the Unicenter CA-Explore for CICS libraries (`expc.library`).
3. Replace `DISKxxx` with the filename of your disk log file.
4. After the `INPUT` command, insert your history reporting commands.

Specifying Report Output Preferences

You can add parameters to your `EXEC` statement to perform the following tasks:

- Suppress the introductory report banner.
- Force all output to upper case.
- Print only lines containing report data.
- Set the date format.
- Specify whether to list report commands before printing the report.

Parameters

Parameter	Purpose
BANNER	Produces a banner that precedes reports and identifies them as output from the Unicenter CA-Explore for CICS report writer. Specify 'BANNER=YES' to include this banner.
CASE	Specify 'CASE=UPPER' to force all output to upper case.
DATAONLY	Specify 'DATAONLY=YES' to print only lines containing report data. If omitted, all lines are printed.
DATEFRMT	Sets the format of dates in report output to <i>mm/dd/yy</i> or <i>dd/mm/yy</i> .
ECHO	Specifies whether to list your report commands before printing the report. Set the value to NO only after you verify the report commands are correct.

Format

```
// EXEC EXPRPT,SIZE=EXPRPT,PARM=[ 'BANNER=NO|YES' ]  
                                [ 'CASE=UPPER' ]  
                                [ 'DATAONLY=[NO|YES]' ]  
                                [ 'DATEFRMT=[MM/DD/YY|DD/MM/YY]' ]  
                                [ 'ECHO=[YES|NO]' ]
```

When using more than one parameter, separate each with a comma:

Step 2: Write History Reporting Commands

The report writer uses two types of history reporting commands:

Global commands – Control the report writer, not any specific report. Global commands, for example, can control the page size the report writer prints, or how many columns the report writer reads in one statement. Global commands can affect all reports included in one execution.

Report commands – Control what happens in a single report. Report commands, for example, can control the time interval or format of a single report.

Global Commands

Global commands set values that stay in effect for all reports printed in one run. If, for example, you use the `LINECOUNT` command to tell the report writer how many lines are on one page, that number of lines applies to all reports in that run.

You can insert global commands anywhere among the report commands. With the exception of the `GLOBAL` command, global commands affect all reports in the run without regard to placement. The `GLOBAL` command affects only reports that come after it.

See the chapter titled “Commands” for a complete list of global commands.

Report Commands

Report commands build a report by telling the report writer how to format the report, what options to use, when to start the report, and so on. The two types of report commands are *primary* and *secondary*.

Primary Report Commands

Each report must contain one (and only one) of the following primary report commands:

TAB—Creates a tabular report showing the value of the specified variables for each time or identifier.

PLOT—Plots the value of a variable horizontally over time.

VPLOT—Plots the value of a variable vertically over time using bars.

MPLOT—Plots the value of a variable vertically over time using outlining.

GRAF—Graphs resource usage by identifier (transaction, terminal, and so on).

FLASHBACK—Produces a report containing one line or page per transaction.

INVOICE—Produces a tabular invoice report using the cost factors you specify.

CANRPT—Produces a canned report.

See the chapter titled “Commands” for a complete list of primary commands.

Secondary Report Commands

After primary report commands, you can include secondary report commands. Secondary report commands alter or enhance the report. You can, for example, use the secondary commands FROM and TO, to limit the time covered by a report, or use TITLE1 to define a title for the report.

See the chapter titled “Commands” for a complete list of secondary commands.

Required Report Commands: END and RUN

Each report must end with the END command, which tells the report writer that the report description is complete.

After the END command, you can start another report. For each new report, the variables you set in the previous report are no longer in effect. If you want to use the same variables, you must enter them again. Values set by global commands, however, remain in effect unless overridden by another command.

Use the RUN command to start executing the reports you described. When the report writer encounters RUN, it stops reading your commands and starts writing reports. Because commands after the RUN command are ignored, make sure RUN is always the last command given to the report writer.

Writing One Report

The following example shows the minimum commands necessary to write one report:

```
primarycommand [variable]
END
RUN
```

The commands in this example perform the following functions:

Command	Function																						
<i>primarycommand</i>	Is one of the following primary report commands, which may or may not be followed by one or more report writer variables:																						
	<table border="1"> <thead> <tr> <th>Command</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>TAB</td> <td>Creates a tabular report showing the value of the specified variable for each time or identifier.</td> </tr> <tr> <td>PLOT</td> <td>Plots the value of the specified variable horizontally over time.</td> </tr> <tr> <td>VLOT</td> <td>Plots the value of the specified variable vertically over time using bars.</td> </tr> <tr> <td>MPLOT</td> <td>Plots the value of the specified variable vertically over time using outlining.</td> </tr> <tr> <td>GRAF</td> <td>Graphs resource usage by identifier (transaction, terminal, user, and so on).</td> </tr> <tr> <td>FLASHBACK</td> <td>Produces a report containing one line or page per transaction, according to the options specified.</td> </tr> <tr> <td>INVOICE</td> <td>Produces a tabular invoice report using the cost factors you specify.</td> </tr> <tr> <td>CANRPT</td> <td>Produces a canned report.</td> </tr> <tr> <td>END</td> <td>Indicates the end of this report.</td> </tr> <tr> <td>RUN</td> <td>Generates all of the reports you specified. In this example, only one report was requested.</td> </tr> </tbody> </table>	Command	Description	TAB	Creates a tabular report showing the value of the specified variable for each time or identifier.	PLOT	Plots the value of the specified variable horizontally over time.	VLOT	Plots the value of the specified variable vertically over time using bars.	MPLOT	Plots the value of the specified variable vertically over time using outlining.	GRAF	Graphs resource usage by identifier (transaction, terminal, user, and so on).	FLASHBACK	Produces a report containing one line or page per transaction, according to the options specified.	INVOICE	Produces a tabular invoice report using the cost factors you specify.	CANRPT	Produces a canned report.	END	Indicates the end of this report.	RUN	Generates all of the reports you specified. In this example, only one report was requested.
Command	Description																						
TAB	Creates a tabular report showing the value of the specified variable for each time or identifier.																						
PLOT	Plots the value of the specified variable horizontally over time.																						
VLOT	Plots the value of the specified variable vertically over time using bars.																						
MPLOT	Plots the value of the specified variable vertically over time using outlining.																						
GRAF	Graphs resource usage by identifier (transaction, terminal, user, and so on).																						
FLASHBACK	Produces a report containing one line or page per transaction, according to the options specified.																						
INVOICE	Produces a tabular invoice report using the cost factors you specify.																						
CANRPT	Produces a canned report.																						
END	Indicates the end of this report.																						
RUN	Generates all of the reports you specified. In this example, only one report was requested.																						

Writing Multiple Reports

To write multiple reports in a single run, end each set of report commands with the END command. Specify the RUN command following the END command for the last report.

Each flashback report must be run as a single report in a separate run. In other words, you cannot specify another primary report command (such as TAB or PLOT) in a job stream with the FLASHBACK command.

The following example shows the minimum commands needed to write **two** reports: a GRAF report and a PLOT report. Note that each report includes an END command.

```
GRAF variable
END

PLOT variable
END

RUN
```

Tailoring Reports Using Secondary Commands

You can tailor a report by inserting secondary commands before the END command in the report.

You can, for example, add the SHIFT command to include in the graph report only the hours from 8:00 a.m. to noon, as follows:

```
GRAF variable
SHIFT 08:00:00 12:00:00
END

PLOT variable
END

RUN
```

Because the SHIFT command is not used with the PLOT command, the resulting plot report will cover all 24 hours. To apply the same time restrictions to the plot report as to the graph report, include the SHIFT command with the PLOT command as well as with the GRAF command.

You can use the same secondary command for each report you create. If, for example, you are printing 10 reports, you can use the SHIFT command with each primary report command to specify the period you want each report to cover. Generally, you cannot use the same secondary command twice for the same report, even if you specify two different time periods; the exceptions to this rule are the COST, EXCLUDE, GROUP, INCLUDE, and RANGE commands, which can be repeated in the same report.

The following figure shows an incorrect use of secondary commands in a command stream and the corrected command stream:

Incorrect Command Stream	Corrected Command Stream
GRAF variable SHIFT 08:00:00 17:00:00 END	GRAF variable SHIFT 8:00:00 17:00:00 END
PLOT variable END RUN	GRAF variable SHIFT 13:00:00 17:00:00 END PLOT variable END RUN

Note: The SHIFT commands in this example are indented to make each list of commands more readable. A command can begin anywhere, as long as it is the leftmost word on the line.

Step 3: Incorporate Report Commands into the Job Stream

After you have written your history reporting commands, insert them into the job stream you defined in Step 1.

Assume, for example, that you have written the following report commands to create a VPLOT report that shows transaction use for each half hour in a 24-hour period:

```
VPLOT TRANSACTION USE
  FROM 01/03/00 00:00:00
  TO   01/03/00 23:59:59
  EACH 30 MINUTES
END
RUN
```

Insert this set of report commands into the job stream as shown below. (This job stream uses input from the flashback file, as explained earlier in this chapter.)

```
* $$ JOB
* $$ LST CLASS=A
// JOB FBCKRPT      PERFORMANCE MONITOR REPORT WRITER
// DLBL EXPCFBK, 'EXPCFBK.FILE', ,VSAM
// LIBDEF *,SEARCH=expc.library
// EXEC EXPRPT,SIZE=EXPRPT
INPUT(EXPCFBK)
*****
VPLOT TRANSACTION USE
  FROM 01/03/00 00:00:00
  TO   01/03/00 23:59:59
  EACH 30 MINUTES
END
RUN
*****
/*
/&
* $$ EOJ
```

Reading Reports

This chapter explains how to read the four types of report writer reports:

- **Tabular reports** – Tabular reports are composed of rows and columns that show the values of variables for one or more individual resources or times.
- **Plot reports** – Plot reports plot the value of a variable for a group of resources either against time or against another variable.
- **Graph reports** – Graph reports are graphs used to compare the value of variables for several resources of the same type.
- **Flashback reports** – Flashback reports are tabular reports that contain detailed information about transactions.

Reading Tabular Reports

You can create the following types of tabular reports using either the TAB command alone or the TAB command together with the TAB2 command:

- Reports generated by the TAB command, which may be either simple or complex:
 - Simple tabular reports give information about a single group of resources and are organized either by resource or by time.
 - More complex tabular reports give information about a single group of resources and are organized by both resource and time period.
- Reports generated by the TAB and TAB2 commands used together, which give information about two groups of related resources

In all tabular reports, a column is created for each variable you specify in the TAB and TAB2 commands. Each report writer variable is assigned an abbreviation that is used by default to create the column heading for that variable's data. To find the meaning of a default heading in a report, refer to the lists of variables in the chapter titled "Variables."

See the chapter titled "Tabular Reports" for a description of how you can change the column headings in a report by using the HDR1 and HDR2 operands.

Simple Tabular Reports

Simple tabular reports are created by the TAB command and are organized either by resource or by time.

In the tabular report structure, the first column of the report lists the key for each row. These keys can be dates and times, or individual resources of a single type, such as terminal IDs, transaction IDs, or operator names.

The following report gives information about several terminals, and is organized by terminal ID. The IDs of all terminals meeting the criteria specified by the report commands are listed in the first column.

TERM ID	TRAN USE	TRAN LIFETIME	TRAN CPU TIME
L203	19	0.318	0.019
L704	16	0.248	0.013
L403	11	1.510	0.050
SUMMARY	46	0.570	0.024

For each period or resource listed in the first column, the remaining columns show the value of each variable specified in the TAB command.

- For tabular reports organized by time or date, the numbers in the body of the report represent the performance of **all** resources of the relevant type during the specified period.
- For tabular reports organized by resource, the rows in the report give the performance of the **one** individual resource whose ID is listed in the first column.

For example, the preceding report shows that terminal L203 invoked 19 transactions, which had an average lifetime of 0.318 seconds and used an average of 0.019 seconds of CPU time.

- The last row in the table is labeled SUM or SUMMARY (depending on the width of the column), and shows the total or average value of each variable for all resources listed. The preceding report, for example, shows that the terminals listed invoked 46 transactions, which lasted an average of 0.570 seconds, and used an average of 0.024 seconds of CPU.

Complex Tabular Reports

Complex tabular reports created by the TAB command are organized by both time and resource; they show the values of variables for specified resources during each time.

The following report is organized by time, and by transaction ID within each time:

TIME	TRAN ID	TRAN USE	TERM RESPTIME	FILE USE
08.00	CINQ	125	0.550	866
	IINQ	82	0.512	730
	PINQ	68	0.297	520
	CMAS	66	1.435	477
	GPCD	27	0.589	224
	SUMMARY	368	0.656	2817
09.00	CINQ	315	0.535	3803
	CMAS	274	1.059	1804
	PINQ	174	0.418	1295
	CENV	76	0.354	107
	INVE	72	0.831	210
	SUMMARY	911	0.679	7219

Tabular reports like the preceding report have the following structure:

- The first column lists dates or times; in the preceding report, the first column organizes the report into two periods of one hour each.
- The second column lists, for each time, the resources specified by the ID option used with the TAB command. In the preceding report, the second column lists all transactions active during each time. Notice that the same transaction ID can be listed more than once in the second column, as is the case with transaction CINQ in the preceding report. This is because a transaction can be active during more than one time.
- The remaining columns show the values of the specified variables for each resource listed for each time. The preceding report shows, for example, that transaction PINQ was used 68 times between 8:00 and 9:00, and that those 68 events had the average terminal response time of 0.297 seconds and made an average of 520 file requests.
- The report includes a SUM or SUMMARY row for each time, which shows the total or average value of each variable for that period for all resources listed.

TAB2 Reports

Reports generated by the TAB and TAB2 commands used together show not only the values of variables for a set of resources, but also the breakdown of those variables.

A TAB command was used to generate the sample report that shows statistics about the transaction use of each of three terminals. Adding a TAB2 command to the TAB command generates a report that also shows those transactions used by each terminal, and statistics about each transaction, as in the following report:

TERM ID	TRAN USE	TRAN LIFETIME	TRAN CPU TIME	TRAN ID	TRAN USE	TRAN LIFETIME	TRAN CPU TIME
L203	19	0.318	0.019	PINQ	16	0.099	0.016
				CINQ	3	1.488	0.035
				SUMMARY	19	0.318	0.019
L704	16	0.248	0.013	SINQ	15	0.144	0.012
				SCOM	1	1.818	0.026
				SUMMARY	16	0.248	0.013
L403	11	1.510	0.050	CMAS	4	1.949	0.055
				CINQ	3	2.653	0.083
				PINQ	3	0.169	0.019
				GPCD	1	0.346	0.027
				SUMMARY	11	1.510	0.050
SUMMARY	46	0.570	0.024		46	0.570	0.024

The first four columns of the preceding TAB2 report were generated by the TAB command and have the structure explained earlier in this chapter. The columns to the right of these were generated by the TAB2 command and have the following structure:

- The TRAN ID column lists the transaction IDs of all transactions used by each terminal listed in the first column.
- The remaining columns show transaction use, transaction lifetime, and transaction CPU time for each of these transactions. One such column is created for each variable specified with the TAB2 command, in the order specified.
- The information for each table includes a row labeled SUM or SUMMARY (depending of the width of the column), which gives the average or total value for each variable for all the resources listed. In this case, the SUMMARY lines show the total transaction use, average transaction lifetime, and average transaction CPU time for all transactions used by each terminal.
- The last row in the report, also labeled SUM or SUMMARY, gives the total or average for each column for the resources listed.

Horizontal plot reports have the following structure:

- The three lines in the upper left of the report give information about the period covered by the report. You can create plots covering different shifts or periods using the SHIFT command or the PERIOD command. The time represented by each detail line is determined by the EACH command. The default is one day.
- The next line of the report displays the plot's scale. This scale is displayed again just below the plot. You can alter a plot's scale with the SCALE operand.
- The leftmost columns of the report list the dates and times for each detail line. A date is not shown unless it has changed from the preceding line. In the preceding report, each detail line represents an hour.
- The rightmost columns of the report list the numeric values for each detail line. The headings of these columns are the characters used in the plot; what each character represents is shown at the bottom of the report following the SYMBOLS: label.
- Plot reports can show the value of a variable for a single resource or a group of resources of a given type together with the average or total value for all resources of that type. For example, in the preceding report, the average terminal response time for the group of terminals specified in the PLOT command is plotted with Xs, and the average terminal response time for all terminals is plotted with hyphens (-).
- If you do not specify a particular resource or group of resources, the plot will show the value of the variable for all resources of the type associated with the variable. Two plot characters will still be assigned, as in the preceding report, but only the global character will appear in the plot.
- A report summary is given at the end of the report if you specify the RANGE command, or at the end of each period in the report if you specify the PERIOD command. These lines are labeled RANGE: or PERIOD: and indicate the following information in graph and numeric form:
 - Average performance for the entire report
 - Average performance for the period

Vertical plot reports have the following structure:

- The leftmost column shows the scale of the report and identifies the data being plotted (in this case, TRAN USE, or transaction use). This information is repeated in the rightmost column.
- The lines immediately underneath the plot show the dates and times for the vertical detail lines. The detail lines show the value of the specified variable for each period; the variable represented by the plot character is shown near the bottom of the report following the SYMBOLS: heading.
- The time period represented by each vertical detail line in the report is shown in the upper left corner of the report following the word EACH. You can use the EACH command to specify the time period each detail line in the plot is to represent; otherwise the time period for each detail line is calculated dynamically based on the total time period the plot covers. The default is one day. In this case, each detail line covers five minutes.
- The horizontal line labeled A indicates the average value for the variable during the period covered by the report.

VPLOT2 Command Vertical Reports

Using the VPLOT2 command with the VPLOT command causes a second variable to be plotted on top of the first.

The following VPLOT2 report includes the information in the VPLOT report earlier in this chapter (transaction use, plotted with Xs). However, the following report also plots average terminal response time, using Os.

The structure of VPLOT2 reports is the same as the structure of VPLOT reports, with the following exceptions:

- Since two variables are plotted, two scales are shown. The scale to the left of the plot applies to the VPLOT variable (in this case, TRAN USE), and the scale to the right of the plot applies to the VPLOT2 variable (in this case, TERM RESP).
- Two average lines appear as well, one for each variable plotted. The letter A appears to the left of the line for the VPLOT variable, and appears to the right of the line that indicates the average value for the VPLOT2 variable.
- The line labeled SYMBOLS: at the bottom of the report includes two plot characters instead of one and identifies the variable represented by each.

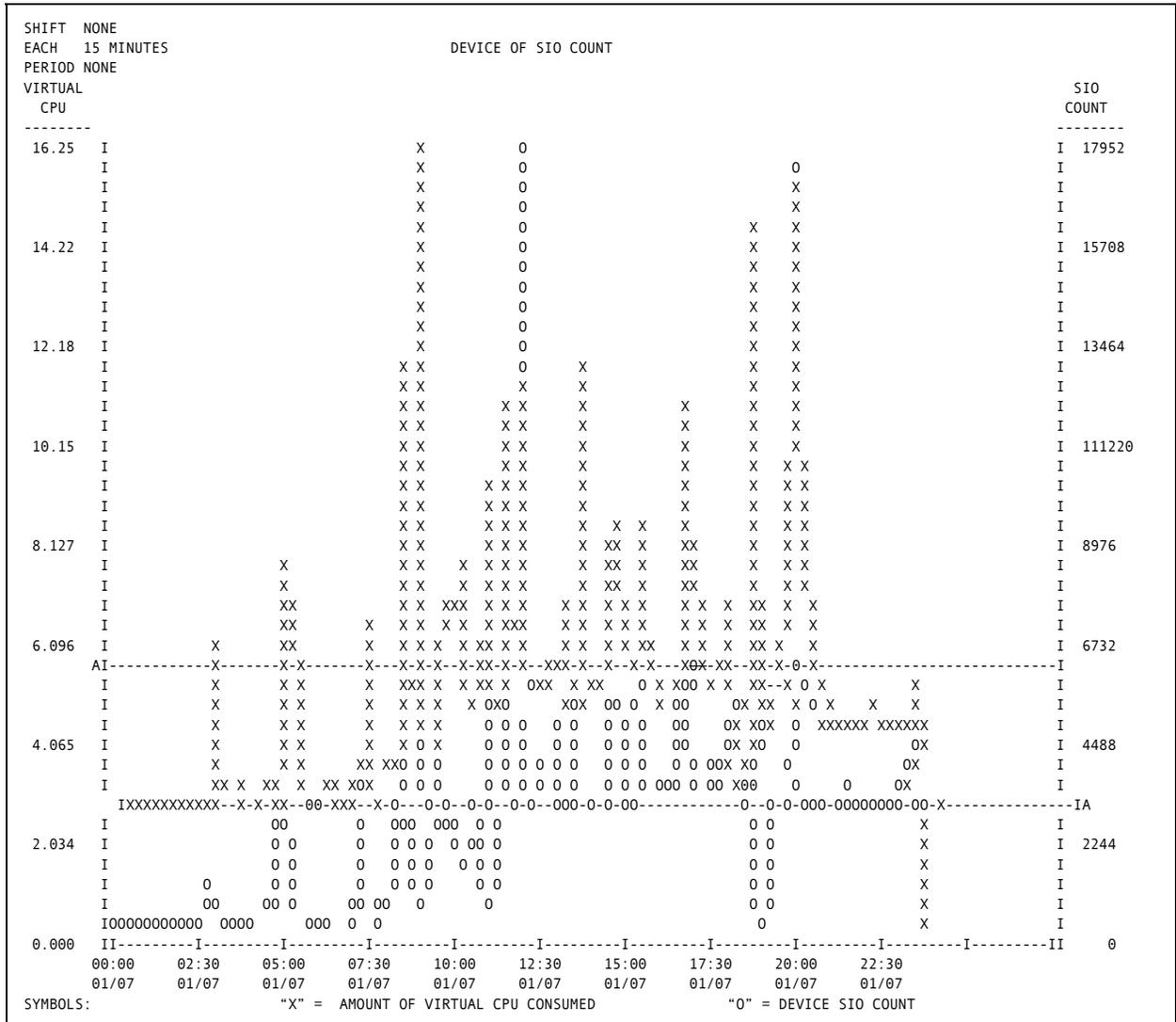
MPLOT Command Vertical Reports

The MPLOT command creates plots similar to those created by the VPLOT or VPLOT2 commands, but only the outline of the plot is shown. Specifying the MPLOT command with a single variable generates a report similar to a VPLOT report; specifying two variables generates a report similar to a VPLOT2 report.

Sample Two-Variable MPlot Report

Note: Two-variable MPlot reports have the same structure as VPLOT2 reports.

The following MPlot report plots the same information as the VPLOT2 report:



Distribution plot reports have the following structure:

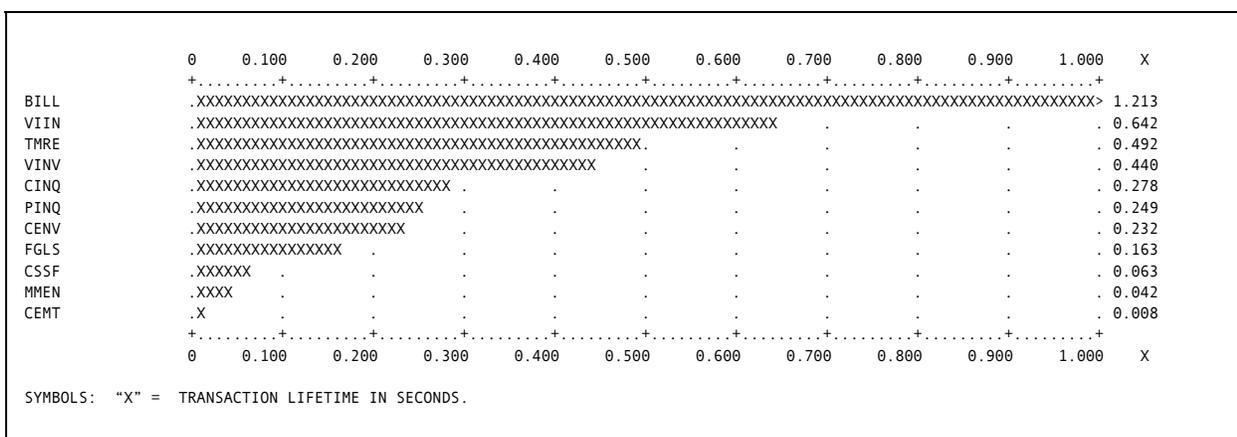
- The leftmost column shows the scale for the VPLOT or MPLOT variable and identifies that variable (in this case TRAN USE).
- The first line immediately underneath the plot shows the scale for the MPLOT variable. This variable is identified by the text that is centered two lines beneath the scale (in this case, TERMINAL AVERAGE RESPONSE TIME, or TERM RESP).
- On the line immediately beneath the scale, the average value of the HPLOT variable is displayed after the label A -. The letter A is positioned at the point on the scale where this average value falls.
- The detail lines in the report show the correlation between the HPLOT variable and the VPLOT or MPLOT variable. If the VPLOT command was specified, the entire detail line is filled in with the plot character; if the MPLOT command was specified, only the outline of the plot is given. If the value of the VPLOT or HPLOT variable exceeds the scale used for that variable, a plus sign (+) appears at the top of the column to indicate the entire column is not included. The preceding report shows that frequently used transactions tend to have low to average terminal response time, while transactions that have high terminal response time tend to be used less frequently.
- If any resources included by the HPLOT variable exceed the scale used for that variable, a detail line is created for those resources; it appears to the right of the HPLOT scale, as shown in the preceding report.
- The number following the label SAMPLE COUNT near the bottom of the report is the total number of samples in the data file. The number and percentage of those samples that fell within in the HPLOT scale is shown after the SAMPLES WITHIN H-SCALE label, and the number and percentage of those samples that fell within in the VPLOT or MPLOT scale is shown after the SAMPLES WITHIN V-SCALE label. The preceding report was created from 643 samples, 637 (99%) of which fell within the HPLOT scale and all of which fell within the VPLOT scale.

Reading Graph Reports

You can create graph reports using either the GRAF command alone, or the GRAF command together with the GRAF2 command.

Graph reports created using the GRAF command alone will graph the value of a single variable for several resources of the same type. For example, a graph report might show the response time of each terminal in a group or the CPU usage of all operators on a system.

The following is a partial graph report showing the lifetimes of several transactions:



Graph reports have the following structure:

- The first line shows the scale used in the report. This scale is also shown just below the plot. You can alter a graph's scale with the SCALE operand of the GRAF command.
- The leftmost column lists the individual resources represented in the graph (in this case, transaction IDs). By default, a graph report includes information about all the individual resources of the type associated with the variable specified. You can override this default organization by specifying an ID option with the GRAF command.

By default, resources are listed in descending order. In this case, the transactions with the longest lifetimes are listed first.
- The rightmost column lists the value that each bar graph represents. The heading of this column is the character used in the graph; what the character represents is shown at the bottom of the report following the SYMBOLS: label.
- If a variable's value exceeds the scale used, a greater-than sign (>) appears in the far right column of the bar graph. This occurs in the preceding report for transaction BILL, which had a lifetime of 1.213 seconds.

Reading Flashback Reports

This topic explains how to read the following types of flashback reports:

- Flashback List reports
- Flashback Detail reports

Flashback List Reports

Flashback List reports contain one line of data for each transaction meeting the criteria specified with the FLASHBACK LIST command.

The following sample is a partial Flashback List report giving information about the transactions used by operator SWS:

CICS ID	END TIME	TERM ID	OPER ID	TRAN ID	TRAN NUMBER	TRAN LIFE	TRAN CPU	TERM RESP	TERM IO	TRAN I/O	FILE COUNT	FILE USE	PROG COUNT	ABEND CODE
B1	00.45.29	0A02	SWS	POSS	67	1.011	0.274	1.011	2	0.000	0	0	3	
B1	00.45.35	0A02	SWS	POSS	69	2.387	0.051	1.187	3	0.199	1	2	6	
B1	00.45.38	0A02	SWS	POSS	70	0.419	0.249	0.000	1	0.000	0	0	3	
B1	00.45.44	0A02	SWS	DEDA	71	5.040	0.705	0.992	4	0.034	1	1	4	
B1	00.45.44	0A02	SWS	POSS	72	0.078	0.023	0.034	2	0.039	1	1	4	
B1	00.46.03	0A02	SWS	POSS	73	0.025	0.002	0.025	2	0.000	0	0	3	
B1	00.46.12	0A02	SWS	SPAM	75	1.422	0.291	0.462	3	0.040	1	1	4	
B1	00.46.18	0A02	SWS	SPAM	76	0.446	0.085	0.199	3	0.052	1	1	4	
B1	00.46.37	0A02	SWS	SPAM	77	0.094	0.020	0.000	1	0.048	1	1	4	

	00.46.37				650	1.213	0.189	0.597	21	0.046	1	7	4	

Flashback List reports have the following structure:

- The first several columns in the report serve to organize or group the transactions being reported on. Flashback List reports can be organized by time and date, by jobname, by CICS region ID, or by any combination of the three. Within this primary organization, a type of resource, such as terminals or operators, can organize the report.

For example, the first column in the preceding report lists, in alphanumeric order, the region IDs of all the transactions included in the report. The second column lists, in chronological order, the transactions used in each region. The third column lists the terminal ID of the terminal that executed each transaction, ordered alphanumerically for each time.

- The remaining columns in the report show the values of various variables for each transaction. The preceding report shows the variables that are included by default. You can include different columns by specifying one or more variables with the FLASHBACK LIST command.

Meanings of Default Column Headings in Flashback Reports

Flashback reports have one or more columns for variable data. Each report writer variable is assigned an abbreviation that is used by default to create the column heading for that variable's data. For explanations of the default headings in flashback reports, refer to the chapter titled "Flashback Reports." To find the meaning of a default heading created by a variable you specify with the FLASHBACK LIST command, refer to the description of the variable in the chapter titled "Variables."

You can change the column headings for variables you specify by using the HDR1 and HDR2 operands of the FLASHBACK LIST command.

Flashback Detail Reports

A Flashback Detail report contains one page of data for each transaction. You can limit a report in a variety of ways. For example, you can report on just those transactions you specify, or just those transactions used by a given operator.

Canned Reports

A *canned report* has commands that are already written for you. The report writer includes canned reports on many frequently requested topics. This chapter explains and describes canned reports furnished by Unicenter CA-Explore for CICS, and includes the following information:

- An explanation of how to produce reports with the CANRPT command
- An explanation of how to create new (customized) canned reports
- A quick reference list of all canned reports
- A description of each canned report and a sample of each report's output

Generating Canned Reports

To generate a canned report, you use the CANRPT command followed by the name of the canned report you want to produce.

When you enter the commands for the canned report of your choice, the report writer generates all the history reporting commands and variables needed for the report.

Command Syntax and Operand

The syntax of the CANRPT command is as follows:

```
CANRPT reportname
```

The operand for the CANRPT command has the following meaning:

Operand	Meaning
<i>reportname</i>	The name of the canned report you want to create

At a minimum, you must use the following commands to run a canned report:

- The CANRPT command followed by the report name
- The END command to end the report
- The RUN command to run the report

To produce a CICSSTAT report, you insert the following commands into a job stream and run the job stream as described in the chapter titled “Writing a Report.”

```
CANRPT CICSSTAT
END
RUN
```

Command Expansions

When the report writer encounters the CANRPT command, it prints on your report the commands that the report writer is generating. This list of commands is called an *expansion*.

The purpose of an expansion is to help you understand how the history reporting commands work. Once the commands make sense, you can create your own reports using the same commands you see in the CANRPT expansions, but tailoring them to meet your needs.

The expansion of the CANRPT CICSSTAT command is as follows:

```
CANRPT CICSSTAT (1)
ERPT237I CANNED REPORT EXPANSION (2)
+*-----* (3)
+* VARIOUS CICS STATISTICS BROKEN DOWN * (3)
+* BY SPECIFIC CICS SYSTEMS. * (3)
+*-----* (3)
+TAB CICSID TRAN USE AND, (4)
+ TRAN USE PCT AND, (4)
+ TRAN USE RATE HDR2(RATE) AND, (4)
+ TRAN LIFE AND, (4)
+ TRAN MAXLIFE AND, (4)
+ TERM RESP AND, (4)
+ TERM MAXRESP AND, (4)
+ TRAN CPUTIME AND, (4)
+ TRAN I/OTIME AND, (4)
+ TRAN WORKSET AND, (4)
+ FILE USE AND, (4)
+ FILE SERVTIME (4)
+ERPT237I END OF CANRPT EXPANSION (5)
secondary commands (6)
```

The following explanations correspond to the numbers in the preceding example:

1. The CANRPT command, which generates the commands in the expansion.
2. A message generated after each CANRPT command to inform you that the following statements were expanded by the report writer.
3. Comments generated by the CANRPT command to document the report.
4. The actual commands that create the report. The CANRPT CICSSTAT command generated these commands.
5. A message that the CANRPT expansion is complete.
6. Any secondary commands that you specified following the CANRPT command appear following the expansion. Secondary commands are not included in the sample expansions in this chapter.

Tailoring Canned Reports

With the few exceptions shown below, you can tailor any canned report by adding secondary commands (such as FROM and TO) between the CANRPT command and the END command.

Secondary commands are described in detail in the chapter titled “Commands.”

The length of reports generated by the CANRPT command depends upon the canned report requested, the activity of your system, and the amount of data in your input file. It is recommended that you use the COUNT command, the SELECT command, or the FROM and TO commands to limit report size.

You cannot add an EACH command to the following canned reports:

- CICS.DLIDBD
- CICS.DLIPGM6
- CICS.DLIPGM7
- CICS.DLIPSB6
- CICS.DLIPSB7
- CICS.FILESTAT
- CICS.TERMSTAT

Although you can add commands after the CANRPT command, you **cannot** reorder the columns in the report or replace the variables in the report.

You **cannot** add a secondary command that is already included in a canned report's expansion.

The following example uses the FROM and TO commands to limit the data in the CICSSTAT report to the week of January 20, 2003:

```
CANRPT CICSSTAT
  FROM 01/20/03
  TO   01/24/03
END
RUN
```

The following example uses the RANGE command to include only terminals with more than 10 output messages in the CICS.TERMSTAT report:

```
CANRPT CICS.TERMSTAT
  RANGE TERM OUTPUTS 10+ DEFERRED
END
RUN
```

The following example uses the COUNT and SHIFT commands to limit the CICS.TERMRESP.RANGE report to 8000 data records for the shift between 8 a.m. and 4 p.m.:

```
CANRPT CICS.TERMRESP.RANGE
  COUNT 8000 SELECTED
  SHIFT 08:00:00 16:00:00
  EACH 1 HOUR
END
RUN
```

The following example uses the SELECT command to limit the CICS.FILESTAT report to files with names beginning with TEST:

```
CANRPT CICS.FILESTAT
  SELECT FILEID(TEST*)
END
RUN
```

Creating Your Own Canned Reports

The report writer uses a report definition table (RDT) to generate canned reports. The default RDT is contained in the phase EXPRDT. The report writer loads the phase EXPRDT upon detection of the first CANRPT command.

You can create customized canned reports in either of the following ways:

- You can add new canned reports to the supplied member EXPRDT.A or change existing reports in it.
- You can create an RDT containing your new canned reports.

Each of these methods is explained further in the following sections.

Method 1: Add Reports to the Supplied Member

To add new canned reports to the supplied member EXPRDT.A or change existing reports contained in it, follow these steps:

Step	Action
1	Edit the member EXPRDT.A in the Unicenter CA-Explore for CICS product <i>library.sublibrary</i> . Add any new canned reports or customize existing reports to suit your needs.
2	Assemble and link edit the member EXPRDT.A.
3	Save the member EXPRDT.A.

WARNING! *If you modify the supplied member EXPRDT.A, make a backup copy of the member using another name. When you install a new version of Unicenter CA-Explore for CICS, EXPRDT.A will be overwritten. Unless you have a backup copy of the member, you will lose any modifications you made.*

To run any canned report in the member EXPRDT.A after you assemble and link edit the member, use the following commands, replacing *your-report-name* with the name of the canned report you want to run:

```
CANRPT your-report-name
END
RUN
```

Method 2: Create an RDT Containing Your New Canned Reports

To create an RDT containing your new canned reports, follow these steps:

Step	Action
1	Create a job stream that uses the ERPTTAB macro to create an RDT. A sample job stream is given in the following section.
2	Decide whether or not to link edit your RDT as EXPRDT, considering the following: <ul style="list-style-type: none"> ■ If you link edit your RDT as EXPRDT, then you should copy the member EXPRDT.A into your job stream. All supplied canned reports and your new canned reports will then be contained in the same member, and you will never have to specify the phase name when you run a canned report. <p>WARNING! If you link edit your RDT as EXPRDT and fail to copy the member EXPRDT.A into your job stream, all canned reports will be lost.</p> ■ If you link edit your RDT using a name other than EXPRDT, then you must specify the phase name each time you run any of your customized reports.
3	Run the job stream to assemble and link edit your RDT.

The following is a sample job stream that assembles and link edits an RDT:

```
* $$ JOB JNM=EXPRDT,CLASS=0,USER='user information'
* $$ LST CLASS=A
// JOB EXPRDT CREATE REPORT DEFINITION TABLE
// OPTION NOXREF
// OPTION SOURCE,SEARCH=expc.library
// OPTION PHASE,CATALOG=expc.library
// OPTION CATAL,NODECK
  PHASE phase-name,* (1)
// EXEC ASSEMBLY
  ERPTTAB your-report-name, (2)
  '-----*', (3)
  '* Add commands in single quotes here. *',
  '* The following lines show sample commands *',
  '-----*',
  'TAB DATETIME TRANID TRANSACTION USE RATE AND,',
  '          TRANSACTION USE AND,',
  '          TRANSACTION LIFE AND,',
  '          DATASET USE AND,',
  '          PROGRAM USE
  END
/*
// EXEC LNKEDT
/&
* $$ EOJ
```

Modify the sample job stream as follows:

1. Replace *phase-name* with the name you want to assign to this RDT phase.
2. Replace *your-report-name* with the name you want to assign to your report, followed by a comma.
3. On the lines following *your-report-name*, insert report commands within single quotes. Follow each command's ending quote with a continuation comma.

Use the following commands to run your report. If you link edited the RDT as EXPRDT, you do not need to specify the parameter *MEMBER=phase-name*.

```
CANRPT your-report-name MEMBER=phase-name
END
RUN
```

Canned Report Descriptions

The following sections describe each of the canned reports supplied with Unicenter CA-Explore for CICS.

The description of each report provides sample commands for running the report, including secondary commands (such as FROM and TO), that are recommended to limit report size. Heading information is omitted from the sample reports.

The following table lists the canned reports provided with Unicenter CA-Explore for CICS:

This Report	Displays
CICSSTAT	CICS statistics broken down by individual CICS systems
CICS.DCT	A graph of transient I/O
CICS.DLIDBD	DL/I activity statistics for each database
CICS.DLIPGM6	DL/I activity statistics (by type of request) made by all programs accessing DL/I data. This CANRPT must be used only on data collected by Unicenter CA-Explore for CICS Release 6.x.

This Report	Displays
CICS.DLIPGM7	DL/I activity statistics (by type of request) made by all programs accessing DL/I data. This CANRPT must be used only on data collected by Unicenter CA-Explore for CICS Release 7.x.
CICS DLIPSB6	DL/I activity statistics (by type of request) made through all PSBs accessing DL/I data. This CANRPT must be used only on data collected by Unicenter CA-Explore for CICS Release 6.x.
CICS DLIPSB7	DL/I activity statistics (by type of request) made through all PSBs accessing DL/I data. This CANRPT must be used only on data collected by Unicenter CA-Explore for CICS Release 7.x.
CICS.FILESTAT	All VSAM datasets and activity statistics for each file
CICS.FILEWAIT	A graph of dataset waits
CICS.INTERVAL.STAT	All active CICS regions and associated system or interval statistics
CICS.INTERVAL.STORAGE. STAT	All active CICS regions and associated storage statistics
CICS.MAXTIME	All active transactions with a complete breakdown of all transaction maximum lifetime components
CICS.OPERSTAT	All active operators with activity statistics for each transaction
CICS.PERFORMANCE.RECORD. ANALYSIS.STAT	How often each component of the performance record is used
CICS.PROGUSE	A graph of program usage
CICS.REVIEW.STAT	A summary of CICS transactions by time extracted from records logged to the EXPCFIL file

This Report	Displays
CICS.STORAGE	All active transactions with various storage statistics
CICS.SYSTEM.THRESHOLD. STAT	All system thresholds that have been triggered
CICS.TEMPSTOR	All active transactions with various temporary storage statistics
CICS.TERMIO	A graph of terminal I/O
CICS.TERMRESP	A graph of terminal response time
CICS.TERMRESP.RANGE	Transaction use broken down into terminal response-time groups
CICS.TERMSTAT	All terminals that were active and statistics for each
CICS.TERMTRAN	A graph of terminal usage and sub-graph of transaction usage for each terminal
CICS.TIMESTAT	All active transactions with a complete breakdown of all transaction lifetime components
CICS.TRANFILE	All active transactions and their associated files
CICS.TRANFIO	A graph of the transaction use with sub-graphs describing file use for all active files for each transaction
CICS.TRANGETM	A graph of transaction GETMAINS
CICS.TRANLIFE.RANGE	Transaction usage broken down into transaction lifetime groups
CICS.TRANPROG	All active transactions and the associated programs
CICS.TRANRESP	A graph of transaction lifetime and sub-graphs of terminal response time for each transaction

This Report	Displays
CICS.TRANSTAT	All active transactions with activity statistics for each transaction
CICS.TRANSTOR	A graph of average transaction storage usage
CICS.TRANTERM	All active transactions and the associated terminals
CICS.TRANTIO	A graph of transaction use and sub-graphs of terminal I/O for each transaction
CICS.TRANUSE	A graph of transaction use
CICS.TRANWAIT	A graph of transaction waits

CICS and VSE Canned Report

The following table lists the available VSE and CICS combination canned report:

This Report	Displays
VSE.CICS.STAT	System VSE and CICS information

CICSSTAT Report

The CICSSTAT report shows various CICS statistics by CICS ID, sorted by CICS system transaction use. The CICS systems with the highest activity are listed first.

The following commands generate a CICSSTAT report:

```
CANRPT CICSSTAT
END
RUN
```

The CANRPT CICSSTAT command expands as follows:

```
CANRPT CICSSTAT
ERPT237I CANNED REPORT EXPANSION
+*-----*
+*  VARIOUS CICS STATISTICS BROKEN DOWN  *
+*  BY SPECIFIC CICS SYSTEMS.           *
+*-----*
+TAB CICSID  TRAN USE          AND,
+          TRAN USE PCT       AND,
+          TRAN USE RATE HDR2(RATE) AND,
+          TRAN LIFE          AND,
+          TRAN MAXLIFE       AND,
+          TERM RESP          AND,
+          TERM MAXRESP       AND,
+          TRAN CPUTIME       AND,
+          TRAN I/OTIME       AND,
+          TRAN WORKSET       AND,
+          FILE USE           AND,
+          FILE SERVTIME
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICSSTAT command generates a report like the following partial report:

SHIFT	NONE											
EACH	DAY											
PERIOD	NONE											
CICS	TRAN	TRAN	TRAN	TRAN	TRAN	TERM	TERM	TRAN	TRAN	TRAN	FILE	FILE
ID	USE	% USE	RATE	LIFE	MAX LIFE	RESP	MAX RESP	CPU	I/O	WORK SET	USE	I/O TIME
B1	6911	79.6	0.04	51.42	28853	0.248	56.39	0.538	0.107	26883	31226	0.024
A1	1773	20.4	0.01	227.7	46487	0.462	43.30	1.002	0.004	9480	1148	0.006
SUM	8684	100.0	0.04	87.43	46487	0.275	56.39	0.633	0.086	23330	32374	0.023

CICS.DLIDBD Report

The CICS.DLIDBD report shows activity for each DL/I database, in decreasing order of database usage.

You cannot specify an EACH command with the CANRPT CICS.DLIDBD command.

The following commands generate a CICS.DLIDBD report:

```
CANRPT CICS.DLIDBD
FROM 12/06/03
TO 12/08/03
END
RUN
```

The CANRPT CICS.DLIDBD command expands as follows:

```
CANRPT CICS.DLIDBD
ERPT237I CANNED REPORT EXPANSION
+*-----*
+*                                           *
+*  SHOW IN TABULAR FORMAT DLI ACTIVITY BY DBD  *
+*                                           *
+*-----*
+SELECT DLITYPE2(^F,^M,^P,^A)
+TAB DLITYPE2 SORT ALPHA DLI USE FORMAT(0)
+TAB2 FILEIDS HDR1(DBD) HDR2(ID) AND,
+  DLI USE AND,
+  DLI SERVTIME AND,
+  DLI SIO AND,
+  DLI SIOTIME AND,
+  DLI MAXI/OT
+ERPT237I END OF UNCAN EXPANSION
```

The CANRPT CICS.DLIDBD command generates a report like the following partial report:

PERIOD	NONE					
FILE	DBD	DL/I	DL/I	DL/I	DL/I	DL/I MAX
TYPE2	ID	USE	SERVTIME	SIOS	SIO TIME	I/O TIME
D	STDCDBL	23				
	STDCX2P	18	0.016	3	0.095	0.263
	STDCX1P	18	0.001	3	0.006	0.008
	STDCDBP			9	0.027	0.184
	STDIDBP			5	0.009	0.014
	SUMMARY	59	0.010	20	0.030	0.263
SUM		59	0.010	20	0.030	0.263

CICS.DLIPGM6 Report

The CICS.DLIPGM6 report shows activity statistics (by type of request) made by all programs accessing DL/I databases, in decreasing order of program usage.

Note: This CANRPT must be used **only** on data collected by Unicenter CA-Explore for CICS Release 6.x. For data collected by Unicenter CA-Explore for CICS Release 7.x, use the CANRPT CICS.DLIPGM7.

You cannot specify an EACH command with the CANRPT CICS.DLIPGM6 command.

The following commands generate a CICS.DLIPGM6 report:

```
CANRPT CICS.DLIPGM6
FROM 12/06/03
TO 12/08/03
END
RUN
```

The CANRPT CICS.DLIPGM6 command expands as follows:

```
CANRPT CICS.DLIPGM6
  ERPT237I CANNED REPORT EXPANSION
+*-----*
+*          *
+*  SHOW IN TABULAR FORMAT DLI ACTIVITY BY DL/I  *
+*  MODULE FOR CA-EXPLORE FOR CICS 6.5X          *
+*          *
+*-----*
+SELECT DLITYPE2(^F,^D,^P,^A)
+TAB DLITYPE2 SORT ALPHA DLI USE FORMAT(0)
+TAB2 FILEIDS HDR1(PROGRAM) HDR2(ID) AND,
+  DLI USE          AND,
+  DLI SERVTIME     AND,
+  DLI SIO          AND,
+  DLI SIOTIME      AND,
+  DLI GU           AND,
+  DLI GN           AND,
+  DLI GNP          AND,
+  DLI GHU          AND,
+  DLI GHN          AND,
+  DLI GHNP         AND,
+  DLI INSERTS      AND,
+  DLI DELETES      AND,
+  DLI REPLACES
+ERPT237I END OF UNCAN EXPANSION
```

The CANRPT CICS.DLIPGM6 command generates a report like the following partial report:

PERIOD	NONE													
FILE	PROGRAM	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I
TYPE2	ID	USE	SERVTIME	SIOS	SIO TIME	GU	GET NEXT	GNP	GHU	GHN	GHNP	INSERT	DELETES	REPLACES
M	DLZHLA60	573		31	0.009	169	400		1			2		1
	DFHTDLI	353	0.001	116	0.004	6	189	39	3	65	26			
	DLZSAM60	254	0.001	14	0.026	73	177		1			2		1
	SUMMARY	1180	0.001	161	0.007	248	766	39	5	65	26	4		2
SUM		1180	0.001	161	0.007	248	766	39	5	65	26	4		2

CICS.DLIPGM7 Report

The CICS.DLIPGM7 report shows activity statistics (by type of request) made by all programs accessing DL/I databases, in decreasing order of program usage.

Note: This CANRPT must be used **only** on data collected by Unicenter CA-Explore for CICS Release 7.x. For data collected by Unicenter CA-Explore for CICS Release 6.x, use the CANRPT CICS.DLIPGM6.

You cannot specify an EACH command with the CANRPT CICS.DLIPGM7 command.

The following commands generate a CICS.DLIPGM7 report:

```
CANRPT CICS.DLIPGM7
FROM 12/06/03
TO 12/08/03
END
RUN
```

The CANRPT CICS.DLIPGM7 command expands as follows:

```

CANRPT CICS.DLIPGM7
ERPT237I CANNED REPORT EXPANSION
+*-----*
+*
+* SHOW IN TABULAR FORMAT DLI ACTIVITY BY DL/I *
+* MODULE FOR CA-EXPLORE FOR CICS 7.0 AND ABOVE *
+*
+*-----*
+SELECT DLITYPE2(^F,^D,^P,^A)
+TAB DLITYPE2 SORT ALPHA DLI USE FORMAT(0)
+TAB2 FILEIDS HDR1(PROGRAM) HDR2(ID) AND,
+ DLI USE AND,
+ DLI SERVTIME AND,
+ DLI MAXI/OT AND,
+ DLI GU AND,
+ DLI GN AND,
+ DLI GNP AND,
+ DLI GHU AND,
+ DLI GHN AND,
+ DLI GHNP AND,
+ DLI INSERTS AND,
+ DLI DELETES AND,
+ DLI REPLACES AND,
+ DLI CHKP
+ERPT237I END OF UNCAN EXPANSION
    
```

The CANRPT CICS.DLIPGM7 command generates a report like the following partial report:

PERIOD	NONE														
FILE	PROGRAM	DL/I	DL/I	DL/I MAX	DL/I	DL/I	DL/I								
TYPE2	ID	USE	SERVTIME	I/O TIME	GU	GET	NEXT	GNP	GHU	GHN	GHNP	INSERT	DELETES	REPLACES	CHKPOINT
M	DLZSAM60	254	0.001	0.263	73	177			1			2		1	
	DFHTDLI	158	0.001	0.184	3	70	21			44	11				9
	SUMMARY	412	0.001	0.263	76	247	21		1	44	11	2		1	9
SUM		412	0.001	0.263	76	247	21		1	44	11	2		1	9

CICS.DLIPSB6 Report

The CICS.DLIPSB6 report shows activity statistics (by type of request) made through all program specification blocks (PSBs) accessing DL/I databases, in decreasing order of PSB usage.

Note: This CANRPT must be used **only** on data collected by Unicenter CA-Explore for CICS Release 6.x. For data collected by Unicenter CA-Explore for CICS Release 7.x, use the CANRPT CICS.DLIPSB7.

You cannot specify an EACH command with the CANRPT CICS.DLIPSB6 command.

The following commands generate a CICS.DLIPSB6 report:

```
CANRPT CICS.DLIPSB6
FROM 12/06/03
TO 12/08/03
END
RUN
```

The CANRPT CICS.DLIPSB6 command expands as follows:

```
CANRPT CICS.DLIPSB6
ERPT237I CANNED REPORT EXPANSION
+*-----*
+*                                           *
+*  SHOW IN TABULAR FORMAT DLI ACTIVITY BY PSB *
+*  FOR CA-EXPLORE FOR CICS 6.5X           *
+*                                           *
+*-----*
+SELECT DLITYPE2(^F,^M,^D,^A)
+TAB DLITYPE2 SORT ALPHA DLI USE FORMAT(0)
+TAB2 FILEIDS HDR1(PSB) HDR2(ID) AND,
+  DLI USE AND,
+  DLI SERVTIME AND,
+  DLI SIO AND,
+  DLI SIOTIME AND,
+  DLI GU AND,
+  DLI GN AND,
+  DLI GNP AND,
+  DLI GHU AND,
+  DLI GHN AND,
+  DLI GHNP AND,
+  DLI INSERTS AND,
+  DLI DELETES AND,
+  DLI REPLACES
+ERPT237I END OF UNCAN EXPANSION
```

The CANRPT CICS.DLIPSB6 command generates a report like the following partial report:

PERIOD	NONE	FILE	PSB	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I
TYPE2	ID	USE	SERVTIME	SIOS	SIO TIME	GU	GET NEXT	GNP	GHU	GHN	GHNP	INSERT	DELETES	REPLACES	
P	STBCUSRP	250		9	0.008	73	177								
	STBCUSUP	162	0.003	11	0.047	3	70	21	1	44	11	2		1	
	SUMMARY	412	0.001	20	0.030	76	247	21	1	44	11	2		1	
SUM		412	0.001	20	0.030	76	247	21	1	44	11	2		1	

CICS.DLIPSB7 Report

The CICS.DLIPSB7 report shows activity statistics (by type of request) made through all program specification blocks (PSBs) accessing DL/I databases, in decreasing order of PSB usage.

Note: This CANRPT must be used **only** on data collected by Unicenter CA-Explore for CICS Release 7.x. For data collected by Unicenter CA-Explore for CICS Release 6.x, use the CANRPT CICS.DLIPSB6.

You cannot specify an EACH command with the CANRPT CICS.DLIPSB7 command.

The following commands generate a CICS.DLIPSB7 report:

```
CANRPT CICS.DLIPSB7
FROM 12/06/03
TO 12/08/03
END
RUN
```

The CANRPT CICS.DLIPSB7 command expands as follows:

```
CANRPT CICS.DLIPSB7
  ERPT237I CANNED REPORT EXPANSION
+*-----*
+*
+* SHOW IN TABULAR FORMAT DLI ACTIVITY BY PSB
+* FOR CA-EXPLORE FOR CICS 7.0 AND ABOVE.
+*
+*-----*
+SELECT DLITYPE2(^F,^M,^D,^A)
+TAB DLITYPE2 SORT ALPHA DLI USE FORMAT(0)
+TAB2 FILEIDS HDR1(PSB) HDR2(ID) AND,
+  DLI USE AND,
+  DLI SERVTIME AND,
+  DLI MAXI/OT AND,
+  DLI GU AND,
+  DLI GN AND,
+  DLI GNP AND,
+  DLI GHU AND,
+  DLI GHN AND,
+  DLI GHNP AND,
+  DLI INSERTS AND,
+  DLI DELETES AND,
+  DLI REPLACES AND,
+  DLI CHKP
+ERPT237I END OF UNCAN EXPANSION
```

The CANRPT CICS.DLIPSB7 command generates a report like the following partial report:

PERIOD	NONE													
FILE	PSB	DL/I	DL/I	DL/I MAX	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I	DL/I
TYPE2	ID	USE	SERVTIME	I/O TIME	GU	GET NEXT	GNP	GHU	GHN	GHNP	INSERT	DELETES	REPLACES	CHKPOINT
P	STBCUSRP	250		0.017	73	177								
	STBCUSUP	162	0.003	0.263	3	70	21	1	44	11	2		1	9
	SUMMARY	412	0.001	0.263	76	247	21	1	44	11	2		1	9
SUM		412	0.001	0.263	76	247	21	1	44	11	2		1	9

CICS.FILESTAT Report

The CICS.FILESTAT report is a tabular report showing how each of the VSAM files in your system performed with respect to several variables. The first column of the report lists the file IDs of all active VSAM files, and the remaining columns show the activity for each file. The report is sorted from the most active file to the least active.

You cannot specify an EACH command with the CANRPT CICS.FILESTAT command.

The following commands generate the CICS.FILESTAT report:

```
CANRPT CICS.FILESTAT
  FROM 12/06/03
  TO   12/08/03
END
RUN
```

The CANRPT CICS.FILESTAT command expands as follows:

```
CANRPT CICS.FILESTAT
ERPT237I CANNED REPORT EXPANSION:
+ *****
+ ***
+ ** SHOW IN TABULAR FORMAT WHAT **
+ ** FILES WERE USED AND HOW **
+ ** EACH FILE PERFORMED. **
+ ***
+ *****
+ TAB FILEID,
+ DATASET USE AND,
+ DATASET UPDATES AND,
+ DATASET READNU AND,
+ DATASET READU AND,
+ DATASET DELETES AND,
+ DATASET ADDS AND,
+ DATASET BROWSES AND,
+ DATASET WAITS AND,
+ DATASET SOS AND,
+ DATASET PSOS AND,
+ DATASET XCL AND,
+ DATASET WBUF AND,
+ DATASET SPLITS
+
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICS.FILESTAT command generates a report like the following partial report:

SHIFT	NONE													
EACH	DAY													
PERIOD	NONE													
FILE	FILE	FILE	FILE	FILE	FILE	FILE	FILE	FILE	FILE	SOS	PSOS	EXCL	WBUF	CI/CA
ID	USE	UPDATE	READNUPD	READUPDT	DELETES	ADDS	BROWSE	WAITS						
FXQFILE	11143	2507	6129	2507	0	0	0	0	0	0	0	0	0	0
C15010	8621	35	0	35	0	0	8551	0	0	0	0	0	0	0
C13010	6663	8	250	33	19	234	6119	32	0	0	0	0	0	32
A96130	3112	0	3102	0	0	0	10	0	0	0	0	0	0	0
VG0010	754	0	272	0	0	0	482	0	0	0	0	0	0	0
B48030	570	0	0	0	0	0	570	0	0	0	0	0	0	0
A96070	297	14	34	14	0	99	136	0	0	0	0	0	0	0
A67370	243	1	0	1	0	12	229	6	0	0	0	0	0	6
E01030	220	0	0	0	0	0	220	0	0	0	0	0	0	0
A96071	181	0	0	0	0	0	181	0	0	0	0	0	0	0
VG0020	136	0	136	0	0	0	0	0	0	0	0	0	0	0
A03020	105	0	105	0	0	0	0	0	0	0	0	0	0	0
CAIRAPS	87	0	33	0	0	0	54	0	0	0	0	0	0	0
C09010	78	0	0	0	1	4	73	0	0	0	0	0	0	0
B48070	70	0	70	0	0	0	0	0	0	0	0	0	0	0
FXCTLFL	25	0	25	0	0	0	0	0	0	0	0	0	0	0
A67210	15	0	15	0	0	0	0	0	0	0	0	0	0	0
A96120	15	2	3	5	3	2	0	0	0	0	0	0	0	0
A67270	14	7	0	7	0	0	0	0	0	0	0	0	0	0
Z01404	8	0	8	0	0	0	0	0	0	0	0	0	0	0
E01070	8	0	8	0	0	0	0	0	0	0	0	0	0	0
FXLOGA	7	3	1	3	0	0	0	0	0	0	0	0	0	0
C09020	2	0	2	0	0	0	0	0	0	0	0	0	0	0
SUMMARY	32374	2577	10193	2605	23	351	16625	38	0	0	0	0	0	38

CICS.INTERVAL.STAT Report

The CICS.INTERVAL.STAT report shows all active CICS regions and associated system or interval statistics.

The following commands generate a CICS.INTERVAL.STAT report:

```
CANRPT CICS.INTERVAL.STAT
  EACH 1 HOUR
  FROM -0 DAYS AT 08.00.00
  END
  RUN
```

The CANRPT CICS.INTERVAL.STAT command expands as follows:

```
CANRPT CICS.INTERVAL.STAT
  ERPT237I CANNED REPORT EXPANSION
+*-----*
+* DISPLAY IN TABULAR FORMAT ALL ACTIVE CICS      *
+* REGIONS AND ASSOCIATED SYSTEM/INTERVAL STATS  *
+*-----*
+TITLE2 CICS SYSTEM/INTERVAL STATISTICS
+TAB DAY DATETIME JOBID,
+  CICS INTERVAL TRAN USE          AND,
+  CICS INTERVAL TRAN USE RATE HDR2(RATE) AND,
+  CICS INTERVAL TRAN LIFE        AND,
+  CICS INTERVAL TRAN CPU         AND,
+  CICS INTERVAL TRAN I/O         AND,
+  CICS INTERVAL REGION CPU       AND,
+  CICS INTERVAL REGION SIO       AND,
+  CICS INTERVAL REGION THRESHOLDS AND,
+  CICS INTERVAL GETVIS %USED     AND,
+  CICS INTERVAL DSA %ALLOCATED
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICS.INTERVAL.STAT command generates a report like the following partial report:

WEEKDAY	DATE	TIME	JOB NAME	TRAN USE	TRAN RATE	TRAN LIFE	TRAN CPU	TRAN I/O	CPU TIME	SIO COUNT	THRSHOLD COUNT	GETVIS %USED	DSA %ALLOC
Thursday	03/11/03	08.00	CICS2	2	0.00	0.048	0.001	0.000	2.080	368	0	14.6	38.2
Thursday	03/11/03	09.00	CICS2	7	0.00	7330	0.006	0.000	2.110	413	1	14.6	37.8
Thursday	03/11/03	10.00	CICS2	1	0.00	0.339	0.002	0.000	2.010	368	0	14.6	37.1
Thursday	03/11/03	11.00	CICS2	1	0.00	0.202	0.001	0.000	2.030	368	0	14.6	37.1
Thursday	03/11/03	12.00	CICS2	1	0.00	0.192	0.002	0.000	2.070	368	0	14.6	37.1
Thursday	03/11/03	13.00	CICS2	1	0.00	0.231	0.002	0.000	2.100	368	0	14.6	37.1
Thursday	03/11/03	14.00	CICS2	1	0.00	0.246	0.002	0.000	2.050	368	0	14.6	37.1
Thursday	03/11/03	15.00	CICS2	1	0.00	0.324	0.002	0.000	2.090	368	0	14.6	37.1
Thursday	03/11/03	16.00	CICS2	2	0.00	0.306	0.002	0.000	1.570	276	0	14.6	37.1

CICS.INTERVAL.STORAGE.STAT Report

The CICS.INTERVAL.STORAGE.STAT report shows all active CICS regions and associated storage statistics.

The following commands generate a CICS.INTERVAL.STORAGE.STAT report:

```
CANRPT CICS . INTERVAL . STORAGE . STAT
      EACH 1 HOUR
      FROM -0 DAYS AT 08.00.00
      END
      RUN
```

The CANRPT CICS.INTERVAL.STORAGE.STAT command expands as follows:

```
CANRPT CICS . INTERVAL . STORAGE . STAT
      ERPT237I CANNED REPORT EXPANSION
+*-----*
+* DISPLAY IN TABULAR FORMAT ALL ACTIVE CICS      *
+* REGIONS AND ASSOCIATED STORAGE STATISTICS     *
+*-----*
+TITLE2 CICS GETVIS STATISTICS
+TAB DAY DATETIME JOBID,
+  CICS INTERVAL DSA %ALLOCATED  AND,
+  CICS INTERVAL DSA COMPRESSIONS AND,
+  CICS INTERVAL GETVIS SIZE     AND,
+  CICS INTERVAL GETVIS %USED    AND,
+  CICS INTERVAL GETVIS %USED MAX,
+  HDR2 (MAX%USED)              AND,
+  CICS INTERVAL GETVIS MAXBLOCK AND,
+  CICS INTERVAL GETVIS 31-SIZE  AND,
+  CICS INTERVAL GETVIS 31-%USED AND,
+  CICS INTERVAL GETVIS 31-%USED MAX,
+  HDR2 (MAX%USED)              AND,
+  CICS INTERVAL GETVIS 31-MAXBLOCK
+ERPT237I END OF CANRPT EXPANSION
```

Note: The variable CICS INTERVAL DSA COMPRESSIONS is valid only when using CICS 2.3. Compression values are not collected for CICS TS. For details, see the CA-Explore for CICS *Online User Guide*.

The CANRPT CICS.INTERVAL.STORAGE.STAT command generates a report like the following partial report:

WEEKDAY	DATE	TIME	JOB NAME	DSA %ALLOC	DSA COMP	GETVIS SIZE	GETVIS %USED	GETVIS MAX%USED	GETVIS MAXBLK	31GETVIS SIZE	31GETVIS %USED	31GETVIS MAX%USED	31GETVIS MAXBLK
Thursday	03/11/03	08.00	CICS2	38.2	0	2116k	14.6	14.6	1780k	2020k	28.0	28.0	1199k
Thursday	03/11/03	09.00	CICS2	37.8	0	2116k	14.6	14.6	1780k	2020k	28.0	28.0	1199k
Thursday	03/11/03	10.00	CICS2	37.1	0	2116k	14.6	14.6	1780k	2020k	28.0	28.0	1199k
Thursday	03/11/03	11.00	CICS2	37.1	0	2116k	14.6	14.6	1780k	2020k	28.0	28.0	1199k
Thursday	03/11/03	12.00	CICS2	37.1	0	2116k	14.6	14.6	1780k	2020k	28.0	28.0	1199k
Thursday	03/11/03	13.00	CICS2	37.1	0	2116k	14.6	14.8	1780k	2020k	28.0	28.0	1199k
Thursday	03/11/03	14.00	CICS2	37.1	0	2116k	14.6	14.6	1780k	2020k	28.0	28.0	1199k
Thursday	03/11/03	15.00	CICS2	37.1	0	2116k	14.6	14.6	1780k	2020k	28.0	28.0	1199k
Thursday	03/11/03	16.00	CICS2	37.1	0	2116k	14.6	14.6	1780k	2020k	28.0	28.0	1199k

CICS.MAXTIME Report

The CICS.MAXTIME report shows transaction maximum lifetime and all of its components by transaction ID, sorted by transaction maximum lifetime. Transactions with the longest maximum lifetimes are listed first.

The following commands generate a CICS.MAXTIME report:

```
CANRPT CICS.MAXTIME
  FROM 12/08/03 12:00:00
  TO   12/08/03 16:00:00
  EACH HOUR
END
RUN
```

The CANRPT CICS.MAXTIME command expands as follows:

```
CANRPT CICS.MAXTIME
ERPT237I CANNED REPORT EXPANSION
+*-----*
+*                                     *
+*  SHOW IN TABULAR FORMAT A DETAILED REPORT OF  *
+*  TRANSACTIONS MAXIMUM LIFETIME BREAKDOWN    *
+*                                     *
+*-----*
+TAB  DATETIME  TRANIDS  TRAN MAXLIFE    AND,
+                                     TRAN MAXUSER  AND,
+                                     TRAN MAXCICS  AND,
+                                     TRAN MAXI/O    AND,
+                                     TRAN MAXWAITECB AND,
+                                     TRAN MAXSUSP   AND,
+                                     TRAN MAXWTR    AND,
+                                     TRAN MAXRESOURCE AND,
+                                     TRAN MAXEXPC   AND,
+                                     TRAN MAXTMIOTIME AND,
+                                     TRAN MAXCPU
+
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICS.MAXTIME command generates a report like the following partial report:

SHIFT NONE															
EACH HOUR															
PERIOD NONE															
DATE	TIME	TRAN ID	TRAN MAX LIFE	TRAN MAX USER	TRAN MAX CICS	TRAN MAX I/O	TRAN MAX WAIT	TRAN MAX SUSP	TRAN MAX WTR	TRAN MAX RSCS	TRAN MAX EXPC	TRAN MAX TMIO	TRAN MAX CPU		
12/08/03	12.00	JJJ	5508	0.000	0.000	0.000	0.095	5508	0.004	0.000	0.003	0.000	0.005		
		TCP	912.8	0.000	0.000	0.000	912.2	0.000	0.000	0.000	0.233	0.000	0.428		
		KCP	912.8	0.000	0.000	0.000	0.000	0.000	912.8	0.000	0.000	0.000	0.451		
		CSSN	7.061	0.000	0.000	0.000	0.000	0.000	0.007	0.000	0.001	6.800	0.023		
		CATD	0.201	0.000	0.000	0.000	0.086	0.000	0.006	0.025	0.002	0.000	0.064		
		PSPI	0.195	0.000	0.000	0.000	0.100	0.000	0.032	0.000	0.001	0.013	0.072		
		POSS	0.141	0.000	0.000	0.093	0.000	0.000	0.012	0.000	0.002	0.001	0.032		
		CSAC	0.025	0.000	0.000	0.000	0.000	0.000	0.003	0.000	0.001	0.000	0.016		
		CSNE	0.013	0.000	0.000	0.000	0.000	0.000	0.005	0.000	0.000	0.000	0.007		
		CSGM	0.012	0.000	0.000	0.000	0.000	0.000	0.000	0.001	0.000	0.000	0.002		
		CSPQ	0.010	0.000	0.000	0.000	0.000	0.000	0.000	0.002	0.000	0.000	0.006		
				SUMMARY	5508	0.000	0.000	0.093	912.2	5508	912.8	0.025	0.233	6.800	0.451
		12/08/03	13.00	KCP	912.5	0.000	0.000	0.000	0.000	0.000	912.5	0.000	0.000	0.000	3.043
TCP	912.5			0.000	0.000	0.000	911.7	0.000	0.000	0.000	0.293	0.000	0.699		
RECE	379.4			0.000	0.000	0.000	0.052	0.000	0.007	0.100	0.004	378.6	0.084		
CSSF	0.767			0.000	0.000	0.000	0.000	0.000	0.005	0.027	0.002	0.000	0.031		
PSPI	0.555			0.000	0.000	0.000	0.114	0.000	0.042	0.000	0.002	0.016	0.330		
POSS	0.092			0.000	0.000	0.050	0.000	0.000	0.020	0.000	0.001	0.002	0.030		
CSPQ	0.042			0.000	0.000	0.000	0.000	0.000	0.034	0.000	0.000	0.000	0.005		
CSAC	0.030			0.000	0.000	0.000	0.000	0.000	0.003	0.000	0.001	0.000	0.021		
CSNE	0.022			0.000	0.000	0.000	0.000	0.000	0.003	0.000	0.001	0.000	0.013		
				SUMMARY	912.5	0.000	0.000	0.050	911.7	0.000	912.5	0.100	0.293	378.6	3.043
12/08/03	14.00	TCP	912.8	0.000	0.000	0.000	912.0	0.000	0.000	0.000	0.234	0.000	0.432		
		KCP	912.8	0.000	0.000	0.000	0.000	0.000	912.8	0.000	0.000	0.000	0.451		
		CSPQ	0.021	0.000	0.000	0.000	0.000	0.000	0.002	0.000	0.000	0.000	0.005		
				SUMMARY	912.8	0.000	0.000	0.000	912.0	0.000	912.8	0.000	0.234	0.000	0.451
12/08/03	15.00	TCP	912.6	0.000	0.000	0.000	911.9	0.000	0.000	0.000	0.233	0.000	0.446		
		KCP	912.6	0.000	0.000	0.000	0.000	0.000	912.6	0.000	0.000	0.000	0.490		
		CSPQ	0.011	0.000	0.000	0.000	0.000	0.000	0.002	0.000	0.000	0.000	0.005		
				SUMMARY	912.6	0.000	0.000	0.000	911.9	0.000	912.6	0.000	0.233	0.000	0.490

CICS.OPERSTAT Report

The CICS.OPERSTAT report shows activity by operator ID, sorted by operator transaction use. This report lists every transaction run by every operator and its corresponding transaction usage. Operators with the highest activity are listed first. The report also includes a sub-table of transactions run by each operator.

Depending on system activity and number of operators, the CICS.OPERSTAT canned report may be a very long report. You should use the FROM and TO commands to limit the size of the report, and a large EACH value is recommended.

The following commands generate a CICS.OPERSTAT report:

```
CANRPT CICS.OPERSTAT
  FROM 12/06/03
  TO   12/07/03
END
RUN
```

The CANRPT CICS.OPERSTAT command expands as follows:

```
CANRPT CICS.OPERSTAT
ERPT237I CANNED REPORT EXPANSION:
+*-----*
+*                                           *
+* SHOW IN TABULAR FORMAT THE TRANSACTIONS USED *
+* BY AN OPERATOR AND THE PERFORMANCE THAT WAS *
+* ACHIEVED. *
+*                                           *
+*-----*
+TAB DATETIME OPERIDS,
+   TRAN USE                               AND,
+   TERM RESP                              AND,
+   TERM MAXRESP                           AND,
+   TRAN LIFE                               AND,
+   TRAN MAXLIFE
+TAB2 TRANID TRAN USE
+
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICS.OPERSTAT command generates a report like the following partial report:

\										
SHIFT NONE										
EACH DAY										
PERIOD NONE										
DATE	TIME	OPER ID	TRAN USE	TERM RESP	TERM MAX RESP	TRAN LIFE	TRAN MAX LIFE	TRAN ID	TRAN USE	
12/06/03	00.00	N/A	59	1.005	12.22	778.8	4529	TCP	22	
								KCP	22	
								CSNE	9	
								VGBS	3	
								END	3	
								SUMMARY	59	
		SUMMARY	59	1.005	12.22	778.8	4529		59	
12/07/03	00.00	N/A	805	0.279	4.397	318.2	43995	X25T	288	
								CSNE	186	
								TCP	97	
								CSPQ	69	
								VGBC	57	
								JJJ	47	
								CHAI	28	
								CATD	16	
								CSPG	12	
								CSKP	5	
								SUMMARY	805	
		SPK	215	0.172	6.137	61.80	2390	POSS	85	
								PSPI	78	
								SPAM	43	
								RECM	6	
								CSSF	2	
								BOEF	1	
								SUMMARY	215	
		SWS	135	0.194	7.922	11.59	699.2	PSPI	74	
								POSS	45	
								SPAM	13	
								POPI	3	
								SUMMARY	135	
		MPC	58	0.044	0.190	10.57	388.0	PSPI	31	
								POSS	26	
								PBLM	1	
								SUMMARY	58	
		BPB	13	0.047	0.150	0.070	0.187	POSS	9	
								PSPI	4	
								SUMMARY	13	
		SUMMARY	1226	0.187	7.922	221.3	43995		1226	

CICS.PERFORMANCE.RECORD.ANALYSIS.STAT Report

The CICS.PERFORMANCE.RECORD.ANALYSIS.STAT report lists all transactions used and shows which components of Unicenter CA-Explore for CICS performance records are being used by each.

You can use the CICS.PERFORMANCE.RECORD.ANALYSIS.STAT report to help determine which CICS resources Unicenter CA-Explore for CICS should collect data about. In particular, if a given component contains a great deal of data, but you seldom run reports on this data, you may choose to stop collecting that data in order to reduce the storage required for the flashback and archive files.

You determine which data Unicenter CA-Explore for CICS collects by setting the values of options in configuration option override members. For information about setting these values and information about the possible implications of changing their default settings, refer to the Unicenter CA-Explore for CICS *Online User Guide*.

For similar tuning information organized by record type rather than by transaction, specify the command `OPTION(RECSTAT=YES)` with any report. Your report will be followed by statistics about Unicenter CA-Explore for CICS performance, accounting, and threshold records. This information includes the length of each record and each of its components (in the case of performance records), the percentage of the total each record or component comprises, and the time covered by each. The `OPTION` command is described in the chapter titled "Commands."

The following commands generate a CICS.PERFORMANCE.RECORD.ANALYSIS.STAT report. The `OPTION` command causes zero values to print as zeros instead of blanks.

```
CANRPT CICS . PERFORMANCE . RECORD . ANALYSIS . STAT
      OPTION(ZEROFLD=ZERO)
      FROM 01/06/03
      TO   01/08/03
END
RUN
```

The CANRPT CICS.PERFORMANCE.RECORD.ANALYSIS.STAT command expands as follows:

```

CANRPT CICS.PERFORMANCE.RECORD.ANALYSIS.STAT
ERPT237I CANNED REPORT EXPANSION
+*-----*
+*  DISPLAY AN ANALYSIS OF THE PERFORMANCE RECORD  *
+*-----*
+TITLE2 PERFORMANCE RECORD ANALYSIS
+TAB DATE TRANID,
+  CICS PERF TRAN USE FORMAT(6N) AND,
+  CICS PERF TRAN RTRHDC,
+  FORMAT(4N) HDR1(HDR) HDR2(CNT) AND,
+  CICS PERF TRAN RTMAINC,
+  FORMAT(4N) HDR1(MAIN) HDR2(CNT) AND,
+  CICS PERF TRAN RTCLCKC,
+  FORMAT(4N) HDR1(CLCK) HDR2(CNT) AND,
+  CICS PERF TRAN RTCNTRC,
+  FORMAT(4N) HDR1(CNTR) HDR2(CNT) AND,
+  CICS PERF TRAN RTSTGTC,
+  FORMAT(4N) HDR1(STG) HDR2(CNT) AND,
+  CICS PERF TRAN RTFILEC,
+  FORMAT(4N) HDR1(FILE) HDR2(CNT) AND,
+  CICS PERF TRAN RTPROGC,
+  FORMAT(4N) HDR1(PGM) HDR2(CNT) AND,
+  CICS PERF TRAN RTTSTGC,
+  FORMAT(4N) HDR1(TSTG) HDR2(CNT) AND,
+  CICS PERF TRAN RTTDATC,
+  FORMAT(4N) HDR1(TDAT) HDR2(CNT) AND,
+  CICS PERF TRAN RTUEHDC,
+  FORMAT(4N) HDR1(EXIT) HDR2(CNT) AND,
+  CICS PERF TRAN RTXSMGC,
+  FORMAT(4N) HDR1(SCTY) HDR2(CNT) AND,
+  CICS PERF TRAN RTJRNLC,
+  FORMAT(4N) HDR1(JRNLC) HDR2(CNT) AND,
+  CICS PERF TRAN RTTHSPC,
+  FORMAT(4N) HDR1(THRS) HDR2(CNT) AND,
+  CICS PERF TRAN RTISCPC,
+  FORMAT(4N) HDR1(ISC) HDR2(CNT) AND,
+  CICS PERF TRAN RTDLISC,
+  FORMAT(4N) HDR1(DL/I) HDR2(CNT) AND,
+  CICS PERF TRAN RTMROCC,
+  FORMAT(4N) HDR1(IRC) HDR2(CNT) AND,
+  CICS PERF TRAN RTCMDLC,
+  FORMAT(4N) HDR1(EIP) HDR2(CNT) AND,
+  CICS PERF TRAN RTDBPIC,
+  FORMAT(4N) HDR1(SQL) HDR2(CNT) AND,
+  CICS PERF TRAN RTTMGPC,
+  FORMAT(4N) HDR1(TABL) HDR2(CNT) AND,
+  CICS PERF TRAN RTERMGC,
+  FORMAT(4N) HDR1(RSCS) HDR2(CNT) AND,
+  CICS PERF TRAN RTEWANC,
+  FORMAT(4N) HDR1(WAIT) HDR2(CNT)
+ERPT237I END OF CANRPT EXPANSION

```

The CANRPT CICS.PERFORMANCE.RECORD.ANALYSIS.STAT command generates a report like the following partial report:

SHIFT NONE																								
EACH DAY																								
PERIOD NONE																								
DATE	TRAN ID	TRAN USE	HDR CNT	MAIN CNT	CLCK CNT	CNTR CNT	STG CNT	FILE CNT	PGM CNT	TSTG CNT	TDAT CNT	EXIT CNT	SCTY CNT	JRNL CNT	THRS CNT	ISC CNT	DL/I CNT	IRC CNT	EIP CNT	SQL CNT	TABL CNT	RSCS CNT	WAIT CNT	
01/06/03	TCP	22	1	1	1	1	1	0	1	0	0	2	0	0	0	0	0	0	0	0	0	1	0	1
	KCP	22	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CSNE	3	1	1	1	1	5	0	3	0	0	2	0	0	0	0	0	0	0	0	4	0	1	
	CATD	2	1	1	1	1	9	0	4	0	1	2	0	2	0	0	0	0	0	4	0	5	0	4
	CSAC	1	1	1	1	1	3	0	1	0	0	3	1	0	0	0	0	0	0	0	0	2	0	2
	CIRB	1	1	1	1	1	9	0	4	1	0	2	1	0	0	0	0	0	0	7	0	3	0	3
	CSDC	1	1	1	1	1	5	0	2	0	1	2	0	0	0	0	18	0	0	0	3	0	5	
	CSGM	1	1	1	1	1	4	0	1	0	0	3	1	0	0	0	0	0	0	0	2	0	2	
	END	1	1	1	1	1	4	0	1	0	0	2	1	0	0	0	0	0	2	0	2	0	2	
	JJJ	1	1	1	1	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
	CSDA	1	1	1	1	1	4	0	1	0	1	2	1	0	0	0	0	0	0	0	3	0	2	
	VGBS	1	1	1	1	1	10	0	8	2	1	3	1	0	0	0	0	0	7	0	4	0	3	
	VGBR	1	1	1	1	1	8	0	2	1	2	2	0	0	0	0	0	0	5	0	4	0	3	
	VGBQ	1	1	1	1	1	9	0	7	5	2	3	1	0	1	0	0	0	6	0	5	0	5	
	SUMMARY	59	1	1	1	1	2	0	1	0	0	1	0	0	0	0	0	0	1	0	1	0	1	
01/07/03	X25T	201	1	1	1	1	10	2	8	7	0	4	1	1	0	0	3	0	7	0	5	2	4	
	PSPI	195	1	1	1	1	8	0	3	1	0	3	1	0	0	0	5	0	5	0	2	2	2	
	CSNE	167	1	1	1	1	5	0	3	0	0	2	0	0	0	0	0	0	0	0	3	0	1	
	POSS	150	1	1	1	1	8	0	4	1	0	3	1	0	0	0	2	0	4	0	3	2	2	
	KCP	74	1	1	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	1	
	TCP	74	1	1	1	1	1	0	0	0	0	3	0	0	0	0	0	0	0	0	1	0	1	
	CSPQ	62	1	1	1	1	4	0	1	0	0	2	0	0	0	0	0	0	0	0	4	0	2	
	VGBC	52	1	1	1	1	8	0	5	4	1	3	1	0	1	0	0	0	6	0	3	0	2	
	JJJ	34	1	1	1	1	0	0	1	0	0	2	0	0	0	0	0	0	0	0	0	0	1	
	SPAM	34	1	1	1	1	9	1	4	0	0	4	1	2	0	0	4	0	4	0	3	2	3	
	CHAI	21	1	1	1	1	6	0	6	0	0	3	1	0	0	0	2	0	3	0	2	2	2	
	PADI	17	1	1	1	1	7	0	5	1	0	3	1	0	0	0	3	0	3	0	3	2	3	
	
	
	
	SUMMARY	1226	1	1	1	1	6	0	4	2	0	3	1	0	0	0	2	0	3	0	3	1	2	
01/08/03	TCP	104	1	1	1	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	1	
	KCP	104	1	1	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	
	PSPI	97	1	1	1	1	7	0	3	1	0	3	1	0	0	0	5	0	4	0	2	2	2	
	CSPQ	52	1	1	1	1	4	0	1	0	0	2	0	0	0	0	0	0	0	0	4	0	2	
	POSS	40	1	1	1	1	8	0	4	1	0	3	1	0	0	0	1	0	4	0	3	2	2	
	CSNE	23	1	1	1	1	5	0	3	0	0	2	0	0	0	0	0	0	0	0	3	0	2	
	CATD	14	1	1	1	1	8	0	4	0	1	2	0	2	0	0	0	0	4	0	5	0	4	
	WISB	14	1	1	1	1	7	0	7	0	0	3	1	0	0	0	2	0	3	0	2	1	2	
	RAPS	12	1	1	1	1	8	0	4	1	0	3	1	0	0	0	0	0	5	0	2	0	2	
	CSGM	6	1	1	1	1	3	0	1	0	0	3	1	0	0	0	0	0	0	0	2	0	1	
	JJJ	6	1	1	1	1	1	0	1	0	0	2	0	0	0	0	0	0	0	0	0	0	1	
	CSSN	5	1	1	1	1	4	0	2	0	1	3	2	0	0	0	0	0	0	0	3	0	2	
	VGBQ	5	1	1	1	1	8	0	5	3	1	2	1	0	0	0	0	0	5	0	4	0	3	
	CSAC	4	1	1	1	1	3	0	1	0	0	3	1	0	0	0	0	0	0	0	2	0	1	
	END	4	1	1	1	1	3	0	1	0	0	2	1	0	0	0	0	0	2	0	2	0	1	
	PINV	4	1	1	1	1	9	0	2	1	0	3	1	0	0	0	2	0	4	0	3	2	4	
	VGBR	4	1	1	1	1	7	0	2	1	2	2	0	0	0	0	0	0	5	0	3	0	2	
	CRSQ	3	1	1	1	1	3	0	1	0	0	1	0	0	0	0	0	0	0	0	3	0	2	
	RECE	3	1	1	1	1	11	1	4	3	0	4	1	1	0	0	1	0	6	0	5	2	5	
	VGBS	3	1	1	1	1	10	0	8	2	1	3	1	0	0	0	0	0	7	0	4	0	3	
	CEMT	2	1	1	1	1	5	0	4	0	0	3	1	0	0	0	0	0	5	0	3	0	2	
	CSFU	2	1	1	1	1	6	0	3	0	0	1	0	0	0	0	0	0	0	0	3	0	2	
	SUMMARY	518	1	1	1	1	4	0	2	0	0	2	0	0	0	0	1	0	2	0	2	1	2	

Field Descriptions

The first two columns following the date list the transactions used and the activity of each.

The next four columns, HDR CNT, MAIN CNT, CLCK CNT, and CNTR CNT, represent transaction and terminal data collected by Unicenter CA-Explore for CICS. These values are always 1 and have no performance tuning implications.

The remaining columns contain information about the storage of various kinds of performance data. The following table lists the variable class each of these columns shows information about:

This Column	Shows Information About This Variable Class
STG CNT	STORAGE
FILE CNT	DATASET
PGM CNT	PROGRAM
TSTG CNT	TEMPSTOR
TDAT CNT	TRANSIENT
EXIT CNT	EXIT
SCTY CNT	SECURITY
JRNL CNT	JOURNAL
THRS CNT	THRESHOLD
ISC CNT	ISC
DL/I CNT	DLI
IRC CNT	IRC
EIP CNT	COMMAND
SQL CNT	SQL
TABL CNT	TABLE
RSCS CNT	RESOURCE
WAIT CNT	WAIT

CICS.REVIEW.STAT Report

The CICS.REVIEW.STAT report shows a summary of CICS transactions by time extracted from records logged to the EXPCFIL file.

The following commands generate a CICS.REVIEW.STAT report:

```
INPUT(EXPCFIL)
CANRPT CICS.REVIEW.STAT
  TO 3/17/03 23.59.59
```

The CANRPT CICS.REVIEW.STAT command expands as follows:

```
INPUT(EXPCFIL)
CANRPT CICS.REVIEW.STAT
  ERPT237I CANNED REPORT EXPANSION
+*-----*
+* DISPLAY IN TABULAR FORMAT ALL INFORMATION      *
+* MAINTAINED IN THE ONLINE EXPCFIL REVIEW RECORD *
+*-----*
+TITLE2 CICS STATISTICAL REVIEW
+TAB DATETIME JOBID,
+  CICS REVIEW TRAN USE          AND,
+  CICS REVIEW TRAN CPU          AND,
+  CICS REVIEW TRAN LIFE         AND,
+  CICS REVIEW TERM RESP         AND,
+  CICS REVIEW FILE SERVTIME     AND,
+  CICS REVIEW FILE REQUESTS     AND,
+  CICS REVIEW FILE SPLITS       AND,
+  CICS REVIEW THRESHOLD COUNT   AND,
+  CICS REVIEW TRAN ABENDS       AND,
+  CICS REVIEW DSA COMP          AND,
+  CICS REVIEW DSA SOS
+ERPT237I END OF CANRPT EXPANSION
END
RUN
```

Note: The CICS REVIEW DSA COMP (compressions) is valid only when using CICS 2.3. Compression values are not collected for TS 1.1. For details, see the Unicenter CA-Explore for CICS *Online User Guide*.

The CICS.REVIEW.STAT command generates a report like the following partial report:

SHIFT	NONE												
EACH	DAY												
PERIOD	NONE												
DATE	TIME	JOB NAME	TRAN USE	TRAN CPU	TRAN LIFE	TERM RESP	FILE I/O TIME	FILE REQS	FILE SPLITS	THRESH COUNT	TRAN ABENDS	DSA COMP	DSA SOS
03/11/03	00.00	CICSAESA	853	0.014	65.29			21					
		CICSBESA	9	0.002	0.003								
		SUMMARY	862	0.013	64.61			21					
03/12/03	00.00	CICSAESA	27	0.030	2.392	0.704	0.001	529		1			
		CICSBESA	16	0.025	8.658	0.034		527		1			
		SUMMARY	43	0.028	4.724	0.455		1056		2			
03/14/03	00.00	CICSAESA	284	0.023	29.51	28.55	0.001	3687		1			
		CICSBESA	9	0.015	10.93	0.071				1			
		SUMMARY	293	0.023	28.94	27.68	0.001	3687		2			
03/15/03	00.00	CICSAESA	1495	0.067	1.211	0.036		56102	12	3582	3570	1	
		CICSBESA	49	0.012	5.493	0.064		527		2			
		SUMMARY	1544	0.066	1.230	0.036		56629	12	3584	3570	1	
03/16/03	00.00	CICSAESA	3425	0.062	28.73	3.506		17939	4	766	765		
		CICSBESA	34	0.024	4.332	0.349		1058		2			
		SUMMARY	3459	0.061	28.49	3.475		18997	4	768	765		
03/23/03	00.00	CICSAESA	764	0.032	2.176	0.059	0.003	111		2			
		SUMMARY	764	0.032	2.176	0.059	0.003	111		2			

CICS.STORAGE Report

The CICS.STORAGE report shows storage activity by transaction ID, sorted by transaction average working set. Transactions with the highest average working set values are listed first.

The following commands generate a CICS.STORAGE report:

```
CANRPT CICS.STORAGE
  FROM 12/06/03
  TO   12/08/03
END
RUN
```

The CANRPT CICS.STORAGE command expands as follows:

```
  CANRPT CICS.STORAGE
  ERPT237I CANNED REPORT EXPANSION
+*-----*
+*                                           *
+*  SHOW IN TABULAR FORMAT A DETAILED REPORT OF  *
+*  TRANSACTION STORAGE USAGE.                *
+*                                           *
+*-----*
+TAB  DATETIME TRANIDS,
+      TRAN WORKSET,
+      TRAN WORKSET MAX HDR1(MAX TRAN)         AND,
+      TRAN STORAGE,                          AND,
+      TRAN GETMAINS,                          AND,
+      TRAN FREEMAINS,                         AND,
+      TRAN GETMSIZE,                          AND,
+      TRAN MAXGETMSIZE
+
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICS.STORAGE command generates a report like the following partial report:

SHIFT NONE		EACH DAY		PERIOD NONE						
DATE	TIME	TRAN ID	TRAN WORK SET	MAX TRAN WORK SET	TRAN STOR	TRAN GETMAINS	TRAN FREEMAIN	TRAN GETMSIZE	TRAN MAX GETM	
12/06/03	00.00	VGBS	36656	36656	194k	594	585	327	4616	
		VGBQ	6088	6088	36384	233	232	156	1864	
		VGBR	3608	3608	4080	15	13	272	1416	
		TCP	3448	17104	10469	79	76	132	5864	
		KCP	1737	16504	9223	2	7	3828	10920	
		SUMMARY	3414	36656	14208	56	56	253	10920	
12/07/03	00.00	DOEF	38829	59088	140k	456	437	307	12088	
		X25T	35482	65496	88588	141	131	627	28488	
		RECM	33499	66840	97888	200	189	489	9240	
		POPI	25480	36408	75416	247	240	305	9288	
		BOEF	22732	34696	63144	145	135	434	9080	
		PBLM	21264	43200	87331	270	258	324	3320	
		WISI	20956	29264	45233	63	59	721	13160	
		RECE	20619	25040	35669	56	47	641	9240	
		PSPI	18975	23072	27464	76	69	360	14280	
		DEDA	15824	15824	21752	34	26	640	6728	
		UPIK	15053	15112	96347	225	212	429	4664	
		PDRN	14008	14008	22784	46	36	495	6376	
		PRNT	7616	7648	112k	236	236	473	3016	
		POSS	7527	9560	9246	22	17	422	3896	
		KCP	5350	16888	47419	14	95	3358	16888	
		VGBC	4016	4016	15212	67	66	226	1864	
		TCP	1759	4792	8792	46	56	189	2056	
		SUMMARY	17239	66840	41004	77	80	530	28488	
12/08/03	00.00	VGBS	95693	127k	590k	2479	2469	238	4616	
		PSPI	18374	20792	21899	34	29	635	14280	
		RECE	18059	20848	28101	43	34	659	9240	
		WISB	12948	29128	26577	23	21	1163	13160	
		POSS	7559	9512	9227	21	17	435	3896	
		VGBQ	4781	7232	20195	97	96	208	1864	
		VGBR	3608	3608	4000	14	13	286	1416	
		KCP	2268	16888	10084	3	13	3319	16888	
		TCP	2267	18896	2355	21	20	114	5864	
		SUMMARY	8334	127k	16392	40	41	406	16888	

CICS.SYSTEM.THRESHOLD.STAT Report

The CICS.SYSTEM.THRESHOLD.STAT report shows all system thresholds that have been triggered, in chronological order.

The following commands generate a CICS.SYSTEM.THRESHOLD.STAT report:

```
CANRPT CICS . SYSTEM . THRESHOLD . STAT
END
RUN
```

The CANRPT CICS.SYSTEM.THRESHOLD.STAT command expands as follows:

```
+*-----*
+* DISPLAY A LIST OF ALL SYSTEM THRESHOLDS *
+* IN CHRONOLOGICAL ORDER *
+*-----*
+TITLE2 CICS SYSTEM THRESHOLD LOG
+TAB ENDTIME FORMAT(0C),
+ CICS SYSTEM THRESHOLD DATE,
+ SORT ASCENDING AND,
+ CICS SYSTEM THRESHOLD TIME XTIME AND,
+ CICS SYSTEM THRESHOLD NAME RIGHT(3) AND,
+ CICS SYSTEM THRESHOLD TYPE AND,
+ CICS SYSTEM THRESHOLD CLASS AND,
+ CICS SYSTEM THRESHOLD RESOURCE AND,
+ CICS SYSTEM THRESHOLD LIMIT RIGHT(3) AND,
+ CICS SYSTEM THRESHOLD VALUE AND,
+ CICS SYSTEM THRESHOLD TERMINAL RIGHT(3) AND,
+ CICS SYSTEM THRESHOLD TRANSACTION AND,
+ CICS SYSTEM THRESHOLD USER
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICS.SYSTEM.THRESHOLD.STAT command generates a report like the following:

DATE	TIME	NAME	TYPE	CLASS	RSCS	THRSHOLD LIMIT	THRSHOLD VALUE
02/18/03	10.49.29.666	CPU%	UPPER	PCT	\$SYSTEMS	90	91
02/18/03	15.03.38.058	CPUTIME	UPPER	TIME	*	1000	15824
02/18/03	15.03.38.058	LIFETIME	UPPER	TIME	*	60000	169K
02/18/03	15.10.59.509	EXPCARC	UPPER	PCT	\$SYSTEMS	90	100
02/18/03	22.27.36.083	LIFETIME	UPPER	TIME	*	60000	26M
02/19/03	22.53.29.589	EXPCARC	UPPER	PCT	\$SYSTEMS	90	100
02/20/03	23.12.59.317	EXPCARC	UPPER	PCT	\$SYSTEMS	90	100

CICS.TEMPSTOR Report

The CICS.TEMPSTOR report shows temporary storage usage and activity by transaction ID, sorted by transaction temporary storage usage. Transactions with the highest temporary storage usage are listed first.

The following commands generate a CICS.TEMPSTOR report:

```
CANRPT CICS.TEMPSTOR
  FROM 12/06/03
  TO   12/08/03
END
RUN
```

The CANRPT CICS.TEMPSTOR command expands as follows:

```
CANRPT CICS.TEMPSTOR
ERPT237I CANNED REPORT EXPANSION
+*-----*
+*          *
+* SHOW IN TABULAR FORMAT A DETAILED REPORT OF *
+* TRANSACTION TEMPORARY STORAGE USAGE.      *
+*          *
+*-----*
+TAB DATETIME TRANIDS ,
+ CICS PERF TEMPSTOR USE AND,
+ CICS PERF TEMPSTOR READS AND,
+ CICS PERF TEMPSTOR PUTQS AND,
+ CICS PERF TEMPSTOR PUTSMAIN AND,
+ CICS PERF TEMPSTOR PUTSAUX AND,
+ CICS PERF TEMPSTOR I/OTIME AND,
+ CICS PERF TEMPSTOR BREQ AND,
+ CICS PERF TEMPSTOR BGET AND,
+ CICS PERF TEMPSTOR BPUT
+
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICS.TEMPSTOR command generates a report like the following partial report:

SHIFT NONE												
EACH DAY												
PERIOD NONE												
DATE	TIME	TRAN ID	TSTG REQS	TSTG READ	TSTG PUTQ	TSTG PUTSMIN	TSTG PUTS AUX	TSTG I/O	TSTG BYTE/REQ	TSTG BYTE/GET	TSTG BYTE/PUT	
12/06/03	00.00	TCP	654	483	166	3	163	0.000	67	68	67	
		VGBS	183	168	7	0	7	0.000	61	63	77	
		VGBQ	89	62	25	0	25	0.000	75	84	60	
		VGBR	3	3	0	0	0	0.000	145	145	0	
		CIRB	2	0	2	0	2	0.000	12	0	12	
SUMMARY			931	716	200	3	197	0.000	67	69	66	
12/07/03	00.00	X25T	5921	4652	980	0	980	0.000	851	557	2501	
		PSPI	2518	95	2257	0	2257	0.000	78	99	83	
		SPON	1379	1046	170	52	118	0.000	354	379	537	
		VGBC	995	833	135	0	135	0.000	88	96	56	
		DOEF	211	14	192	0	192	0.000	126	657	90	
		POSS	112	50	62	0	62	0.000	154	154	154	
		PRNT	74	65	3	3	0	0.000	322	350	350	
		RECM	65	26	25	0	25	0.000	857	1074	1111	
		UPIK	56	44	4	4	0	0.000	300	350	350	
		WISI	28	14	7	0	7	0.000	2051	2805	2596	
		CSPG	22	12	0	0	0	0.000	22	60	0	
		BOEF	22	11	8	0	8	0.000	582	715	619	
		RECE	20	9	6	0	6	0.000	635	847	847	
		RAPP	15	5	0	0	0	0.000	3	8	0	
		PINV	12	8	4	0	4	0.000	154	154	154	
		SPAM	12	6	6	0	6	0.000	154	154	154	
		PADI	10	0	10	0	10	0.000	508	0	508	
		CSPS	6	6	0	0	0	0.000	60	60	0	
		PBLM	5	0	2	0	2	0.000	62	0	154	
		POPI	4	2	2	0	2	0.000	154	154	154	
		PINT	2	1	1	0	1	0.000	154	154	154	
		SOIN	2	1	1	0	1	0.000	154	154	154	
		CHAI	2	0	2	0	2	0.000	592	0	592	
		DEDA	1	1	0	0	0	0.000	154	154	0	
		CSDB	1	1	0	0	0	0.000	0	0	0	
		PART	1	0	0	0	0	0.000	0	0	0	
SUMMARY			11496	6902	3877	59	3818	0.000	526	466	730	
12/08/03	00.00	VGBS	2572	2232	248	0	248	0.000	63	64	77	
		TCP	654	483	166	3	163	0.000	67	68	67	
		PSPI	254	7	157	0	157	0.000	57	154	85	
		VGBQ	149	104	40	0	40	0.000	69	78	54	
		POSS	25	11	14	0	14	0.000	154	154	154	
		RECE	15	7	5	0	5	0.000	609	748	778	
		VGBR	12	12	0	0	0	0.000	145	145	0	
		RAPS	12	6	4	4	0	0.000	1009	1009	1514	
		WISB	9	5	1	0	1	0.000	2009	3013	3013	
		PINV	6	2	2	0	2	0.000	103	154	154	
		CSPQ	3	1	0	0	0	0.033	20	60	0	
		CIRB	2	0	2	0	2	0.000	12	0	12	
		CSDB	1	1	0	0	0	0.000	0	0	0	
SUMMARY			3714	2871	639	7	632	0.033	74	75	96	

CICS.TERMRESP Report

The CICS.TERMRESP report graphs terminal response time (in seconds) for each active terminal.

The following commands generate a CICS.TERMRESP report:

```
CANRPT CICS.TERMRESP
  FROM 01/06/03 08.00.00
  TO   01/06/03 09.00.00
END
RUN
```

The CANRPT CICS.TERMRESP command expands as follows:

```
CANRPT CICS.TERMRESP
ERPT237I CANNED REPORT EXPANSION:
+ *****
+ ***
+ **  GRAF TERMINAL RESPONSE TIME **
+ **  FOR EACH TERMINAL IN THE    **
+ **          SYSTEM.              **
+ ***
+ *****
+ GRAF TERMINAL RESPONSETIME
+
+ERPT237I END OF CANRPT EXPANSION
```


CICS.TERMRESP.RANGE Report

The CICS.TERMRESP.RANGE report shows transaction usage broken down into specific ranges of terminal response time.

The following commands generate a CICS.TERMRESP.RANGE report. The COUNT command causes the input to be limited to 8000 records.

```
CANRPT CICS.TERMRESP.RANGE
COUNT 8000 SELECTED
SHIFT 08:00:00 15:00:00
EACH 1 HOUR
END
RUN
```

The CANRPT CICS.TERMRESP.RANGE command expands as follows:

```
CANRPT CICS.TERMRESP.RANGE
ERPT237I CANNED REPORT EXPANSION:
+*-----*
+* TRANSACTION USE BROKEN DOWN INTO *
+* SPECIFIC TERMINAL RESPONSE GROUPS *
+*-----*
+TAB DATETIME TRAN USE AND,
+TRAN USE RATE HDR2(RATE) AND,
+TRAN USE RANGE(TRESP1),
+ HDR1(TRAN USE) HDR2(0SEC) AND,
+TRAN USE PCT RANGE(TRESP1),
+ HDR1(% USE) HDR2(0-1 SEC) AND,
+TRAN USE RANGE(TRESP3),
+ HDR1(TRAN USE) HDR2(1-3 SEC) AND,
+TRAN USE PCT RANGE(TRESP3),
+ HDR1(% USE) HDR2(1-3 SEC) AND,
+TRAN USE RANGE(TRESP5),
+ HDR1(TRAN USE) HDR2(3-5 SEC) AND,
+TRAN USE PCT RANGE(TRESP5),
+ HDR1(% USE) HDR2(3-5 SEC) AND,
+TRAN USE RANGE(TRESP10),
+ HDR1(TRAN USE) HDR2(5-10 SEC) AND,
+TRAN USE PCT RANGE(TRESP10),
+ HDR1(% USE) HDR2(5-10 SEC) AND,
+TRAN USE RANGE(TRESP+),
+ HDR1(TRAN USE) HDR2(10+ SEC) AND,
+TRAN USE PCT RANGE(TRESP+),
+ HDR1(% USE) HDR2(10+ SEC)
+
+RANGE TERM RESP 0-1 ID TRESP1
+RANGE TERM RESP 1-3 ID TRESP3
+RANGE TERM RESP 3-5 ID TRESP5
+RANGE TERM RESP 5-10 ID TRESP10
+RANGE TERM RESP 10-999 ID TRESP+
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICS.TERMRESP.RANGE command generates a report like the following partial report:

SHIFT NONE		EACH HOUR		PERIOD NONE															
DATE	TIME	TRAN USE	TRAN RATE	TRAN USE 0-1 SEC	% USE 0-1 SEC	TRAN USE 1-3 SEC	% USE 1-3 SEC	TRAN USE 3-5 SEC	% USE 3-5 SEC	TRAN USE 5-10 SEC	% USE 5-10 SEC	TRAN USE 10+ SEC	% USE 10+ SEC						
12/06/03	08.00	26	0.01	25	96.2	0	0.0	1	3.8	0	0.0	0	0.0	0	0.0				
12/06/03	09.00	17	0.00	17	100.0	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0				
12/06/03	10.00	16	0.00	16	100.0	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0				
12/06/03	11.00	48	0.01	45	93.8	3	6.3	0	0.0	0	0.0	0	0.0	0	0.0				
12/06/03	12.00	105	0.03	99	94.3	6	5.7	0	0.0	0	0.0	0	0.0	0	0.0				
12/06/03	13.00	149	0.04	148	99.3	1	0.7	0	0.0	0	0.0	0	0.0	0	0.0				
12/06/03	14.00	151	0.04	151	100.0	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0				
12/07/03	08.00	194	0.05	193	99.5	1	0.5	0	0.0	0	0.0	0	0.0	0	0.0				
12/07/03	09.00	228	0.06	226	99.1	1	0.4	1	0.4	0	0.0	0	0.0	0	0.0				
12/07/03	10.00	92	0.03	91	98.9	1	1.1	0	0.0	0	0.0	0	0.0	0	0.0				
12/07/03	11.00	119	0.03	115	96.6	3	2.5	1	0.8	0	0.0	0	0.0	0	0.0				
12/07/03	12.00	84	0.02	83	98.8	1	1.2	0	0.0	0	0.0	0	0.0	0	0.0				
12/07/03	13.00	56	0.02	54	96.4	0	0.0	2	3.6	0	0.0	0	0.0	0	0.0				
12/07/03	14.00	18	0.01	18	100.0	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0				
12/08/03	08.00	63	0.02	63	100.0	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0				
12/08/03	09.00	39	0.01	39	100.0	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0				
12/08/03	10.00	120	0.03	120	100.0	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0				
12/08/03	11.00	20	0.01	20	100.0	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0				
12/08/03	12.00	20	0.01	20	100.0	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0				
12/08/03	13.00	22	0.01	22	100.0	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0				
12/08/03	14.00	22	0.01	22	100.0	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0				
12/09/03	08.00	44	0.01	44	100.0	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0				
12/09/03	09.00	42	0.01	42	100.0	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0				
12/09/03	10.00	20	0.01	20	100.0	0	0.0	0	0.0	0	0.0	0	0.0	0	0.0				
12/09/03	11.00	88	0.02	80	90.9	4	4.5	1	1.1	1	1.1	2	2.3	2	2.3				
12/09/03	12.00	342	0.10	335	98.0	5	1.5	2	0.6	0	0.0	0	0.0	0	0.0				
12/09/03	13.00	2417	0.67	2342	96.9	61	2.5	10	0.4	2	0.1	2	0.1	2	0.1				
12/09/03	14.00	4122	1.26	3887	94.3	168	4.1	35	0.8	24	0.6	8	0.2	8	0.2				
PERIOD		8684	0.08	8337	96.0	255	2.9	53	0.6	27	0.3	12	0.1	12	0.1				

CICS.TERMSTAT Report

The CICS.TERMSTAT report shows how each active terminal in your system performed with respect to seven variables, sorted by maximum response times, from highest to lowest.

You cannot specify an EACH command with the CANRPT CICS.TERMSTAT command.

The following commands generate a CICS.TERMSTAT report:

```
CANRPT CICS.TERMSTAT
  FROM 12/07/00
  TO   12/08/00
END
RUN
```

The CANRPT CICS.TERMSTAT command expands as follows:

```
CANRPT CICS.TERMSTAT
ERPT237I CANNED REPORT EXPANSION
+ *****
+ ***
+ ** SHOW IN TABULAR FORMAT WHAT **
+ ** TERMINALS WERE USED AND HOW **
+ ** EACH TERMINAL PERFORMED.   **
+ ***
+ *****
+ TAB TERMIDS,
+   TERMINAL MAXRESP           AND,
+   TERMINAL RESP              AND,
+   TERM MESSAGEs             AND,
+   TERM INPUTs               AND,
+   TERM OUTPUTs              AND,
+   TERM INPUTSIZE            AND,
+   TERM OUTPUTSIZE
+
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICS.TERMSTAT command generates a report like the following partial report:

SHIFT	NONE						
EACH	DAY						
PERIOD	NONE						
TERM ID	TERM MAX RESP	TERM RESP	TERM IO	TERM READS	TERM WRITES	TERM BYT-READ	TERM BYT-WRIT
F001	56.39	0.704	381	187	194	2	1507
0A02	7.922	0.194	604	221	383	99	397
0A19	6.137	0.202	1920	719	1201	55	607
CNSL	5.592	1.623	22	11	11	8	100
0A14	4.094	0.265	65	26	39	294	849
8A26	3.510	0.217	64	22	42	11	253
F002	2.900	0.393	37	18	19	97	921
0A12	2.771	0.217	106	51	55	70	382
0A09	1.677	0.064	547	166	381	16	331
X253	1.599	0.371	340	170	170	58	98
X255	0.820	0.407	32	16	16	69	102
X254	0.795	0.292	40	20	20	38	74
8A21	0.767	0.052	346	105	241	14	190
0A0F	0.731	0.172	21	10	11	94	405
X250	0.259	0.060	173	112	61	6	9
0A05	0.229	0.036	89	25	64	11	136
8A09	0.190	0.042	104	33	71	11	223
0A04	0.189	0.038	8	2	6	12	292
4A16	0.021	0.040	2	1	1	4	723
RP3P	0.015	0.006	4	0	4	0	151
RW1P	0.010	0.005	6	0	6	0	559
N/A	0.000	0.000	0	0	0	0	0
SUMMARY	56.39	0.211	4911	1915	2996	49	496

CICS.TERMTRAN Report

The CICS.TERMTRAN report graphs terminal usage for each active terminal. For each terminal, there is also a sub-graph showing what transactions the terminal used and how many times it used each transaction.

The following commands generate a CICS.TERMTRAN report:

```
CANRPT CICS.TERMTRAN
  FROM 12/06/03 08:00:00
  TO   12/06/03 12.00.00
END
RUN
```

The CANRPT CICS.TERMTRAN command expands as follows:

```
      CANRPT CICS.TERMTRAN
ERPT237I CANNED REPORT EXPANSION:
+ *****
+ ***                                     ***
+ **  GRAF TERMINAL USAGE AND          **
+ **  ALL TRANSACTIONS USED FOR        **
+ **  EACH TERMINAL.                   **
+ ***                                     ***
+ *****
+ GRAF  TERM USE
+ GRAF2 TRAN USE
+
+ERPT237I END OF CANRPT EXPANSION
```


CICS.TIMESTAT Report

The CICS.TIMESTAT report shows transaction lifetime and all of its components by transaction ID, sorted by transaction lifetime. Transactions with the longest lifetimes are listed first.

The following commands generate a CICS.TIMESTAT report:

```
CANRPT CICS.TIMESTAT
END
RUN
```

The CANRPT CICS.TIMESTAT command expands as follows:

```
CANRPT CICS.TIMESTAT
ERPT237I CANNED REPORT EXPANSION
+*-----*
+*          *
+*  SHOW IN TABULAR FORMAT A DETAILED REPORT OF  *
+*  TRANSACTION LIFETIME BREAKDOWN              *
+*          *
+*-----*
+TAB  DATETIME  TRANIDS  TRAN LIFETIME    AND,
+          TRAN PGMTIME    AND,
+          TRAN I/OTIME    AND,
+          TRAN WAITTIME   AND,
+          TRAN SUSPTIME   AND,
+          TRAN WTRTIME    AND,
+          TRAN RESOURCETIME AND,
+          TRAN EXPCTIME   AND,
+          TRAN TMIOIME    AND,
+          TRAN CPUTIME
+
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICS.TIMESTAT command generates a report like the following partial report:

SHIFT NONE												
EACH DAY												
PERIOD NONE												
DATE	TIME	TRAN ID	TRAN LIFE	TRAN PGM	TRAN I/O	TRAN WAIT	TRAN SUSP	TRAN WTR	TRAN RSCS	TRAN EXPC	TRAN TMIO	TRAN CPU
03/03/03	00.00	ISQL	7356	3.853		22.23		1.779	0.492	0.078	7328	0.209
		CISQ	7353	2.065		7350		0.036	0.473	0.072		0.180
		IS\$Q	4627	0.001		4627		0.002				0.001
		KCP	304.0							0.024		0.054
		TCP	304.0	0.071		304.0				0.023		0.034
		S140	30.00	0.005			30.00	0.002				0.004
		CSFU	3.253	0.742		2.511				0.001		0.122
		CIRB	2.787	0.734		1.522		0.003	0.525	0.001	0.002	0.031
		IS\$5	1.644	0.003		1.638		0.003				0.002
		IESX	0.913	0.027			0.859	0.027		0.002		0.004
		alxp	0.734	0.726				0.002		0.002	0.006	0.016
		CSSN	0.660	0.448	0.049		0.137	0.008	0.017	0.007	0.002	0.034
		IES6	0.597	0.591	0.002			0.003		0.001	0.001	0.018
		CATD	0.567	0.500		0.046		0.007	0.015	0.002		0.020
		USER	0.204	0.180	0.012			0.003	0.009			0.011
		ALXP	0.177	0.106	0.038	0.020		0.012	0.002	0.011		0.050
		CSNE	0.167	0.157				0.001	0.009	0.001		0.004
		CSAC	0.150	0.148				0.001			0.001	0.003
		CSPQ	0.116	0.104				0.012		0.004		0.002
		IEGM	0.114	0.053				0.001	0.004		0.056	0.004
		CRSQ	0.047	0.047				0.001				0.001
		PF3	0.038	0.032	0.006			0.001		0.001		0.007
		IESI	0.036	0.027	0.009							0.007
		scty	0.031	0.030				0.001			0.001	0.004
		IS\$P	0.018	0.009		0.008		0.001	0.001	0.001		0.006
		IE\$1	0.009	0.008	0.001			0.001		0.001		0.005
		IS\$3	0.008	0.007				0.001				0.003
		IE\$2	0.005	0.004				0.001				0.004
		SCTY	0.005	0.003				0.001			0.001	0.003
		IS\$1	0.004	0.004								0.003
		CSGM	0.004	0.003								0.003
		SUMMARY	117.0	0.038	0.002	54.20	12.45	0.004	0.002	0.008	5.690	0.018
03/04/03	00.00	TCP	304.0	0.046		304.0				0.014		0.035
		KCP	304.0							0.009		0.054
		ALXP	1.004	0.564	0.107	0.194		0.045	0.093	0.044		0.317
		CATD	0.791	0.721		0.058		0.001	0.011	0.002		0.029
		CSTA	0.650	0.591				0.001	0.059			0.014
		IESX	0.618	0.009			0.606	0.003				0.003
		CSSN	0.415	0.318	0.043			0.003	0.050	0.005	0.001	0.038
		PF3	0.196	0.153	0.003			0.001	0.039	0.001		0.013
		XCOM	0.142	0.099	0.019				0.023	0.001		0.009
		IEGM	0.106	0.019				0.001	0.006		0.079	0.003
		CRSQ	0.101	0.101				0.001				0.001
		CSPQ	0.068	0.057				0.001	0.010	0.005		0.002
		alxp	0.067	0.047				0.001	0.019			0.005
		CSNE	0.023	0.016					0.006			0.002
		IESI	0.006	0.006								0.003
		SCTY	0.005	0.004				0.001			0.001	0.003
		CSGM	0.004	0.003				0.001				0.003
		SUMMARY	208.1	0.046	0.003	104.0	0.060	0.001	0.005	0.009	0.002	0.038

CICS.TRANFILE Report

The CICS.TRANFILE report displays information about all active transactions and their associated files, sorted first by transaction use and then by dataset use. Transactions with the highest activity are listed first.

The following commands generate a CICS.TRANFILE report:

```
CANRPT CICS.TRANFILE
  FROM 12/06/00
  TO   12/07/00
END
RUN
```

The CANRPT CICS.TRANFILE command expands as follows:

```
CANRPT CICS.TRANFILE
ERPT237I CANNED REPORT EXPANSION
+*-----*
+* DISPLAY IN TABULAR FORMAT ALL ACTIVE *
+* TRANSACTION AND ASSOCIATED FILES *
+*-----*
+TAB  TRANIDS  TRAN USE      AND,
+          TRAN I/OTIME    AND,
+          TRAN MAXI/OTIME AND,
+          FILE USE        AND,
+          FILE READS      AND,
+          FILE WRITES     AND,
+          FILE SERVTIME
+TAB2 FILEIDS  FILE USE      AND,
+          FILE READS      AND,
+          FILE WRITES     AND,
+          FILE SERVTIME   AND,
+          FILE MAXI/OTIME
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICS.TRANFILE command generates a report like the following partial report:

SHIFT	NONE												
EACH	DAY												
PERIOD	NONE												
TRAN ID	TRAN USE	TRAN I/O	TRAN MAX I/O	FILE USE	FILE READS	FILE WRITES	FILE I/O TIME	FILE ID	FILE USE	FILE READS	FILE WRITES	FILE I/O TIME	FILE MAX I/O TIME
FXFR	2482	0.148	23.64	11172	8662	2510	0.033	FXQFILE	11141	8634	2507	0.033	2.874
								FXCTLFL	24	24	0	0.039	0.321
								FXLOGA	7	4	3	0.604	3.984
								SUMMARY	11172	8662	2510	0.033	3.984
PSPI	939	0.003	0.376	29	29	0	0.091	A96130	29	29	0	0.091	0.188
CSNE	699	0.000	0.000	0	0	0	0.000						
POSS	610	0.050	1.943	323	323	0	0.095	A96130	323	323	0	0.095	1.772
X25T	546	0.253	5.885	8032	7771	261	0.017	C13010	6663	6402	261	0.009	0.744
								A96130	961	961	0	0.077	1.767
								VG0010	272	272	0	0.014	0.275
								VG0020	136	136	0	0.013	0.500
								SUMMARY	8032	7771	261	0.017	1.767
RAPL	474	0.000	0.000	0	0	0	0.000						
WISB	313	0.007	0.468	41	21	20	0.051	A67370	15	2	13	0.123	0.068
								A67270	14	7	7	0.014	0.032
								A67210	12	12	0	0.004	0.028
								SUMMARY	41	21	20	0.051	0.068
WISI	263	0.008	0.207	230	230	0	0.009	A67370	228	228	0	0.009	0.047
								A67210	2	2	0	0.016	0.031
								SUMMARY	230	230	0	0.009	0.047
TCP	224	0.003	0.381	482	482	0	0.001	VG0010	482	482	0	0.001	0.063
KCP	224	0.000	0.000	0	0	0	0.000						
SUMMARY	6774	0.080	0.000	20309	17518	2791	0.027		32374	29423	2951	0.023	3.984

CICS.TRANFIO Report

The CICS.TRANFIO report graphs the number of times each transaction was used together with a sub-graph showing all the VSAM datasets that each transaction used and the number of I/Os to each dataset.

The CICS.TRANFIO report may be very long. To shorten the report, you can do one of the following:

- Limit the time covered by using the FROM, TO, or SHIFT commands with the CANRPT command.
- Use the SELECT command to specify the transactions or datasets to be included in or excluded from the report.
- Use the commands in the report's expansion rather than the CANRPT command, and limit the number of transactions or datasets shown in the report by adding operands to the GRAF and GRAF2 commands. The operands of these commands are explained in the chapter titled "Graph Reports."

The following commands generate a CICS.TRANFIO report:

```
CANRPT CICS.TRANFIO
      FROM 01/06/03 08.00.00
      TO   01/06/03 09.00.00
END
RUN
```

The CANRPT CICS.TRANFIO command expands as follows:

```
CANRPT CICS.TRANFIO
ERPT237I CANNED REPORT EXPANSION:
+ *****
+ ***
+ ** BREAK DOWN TRANSACTIONS BY **
+ ** THEIR USE AND SHOW DATASET **
+ ** USE FOR EACH TRANSACTION, **
+ ** FOR EACH DATASET. **
+ ***
+ *****
+ GRAF TRANSACTION USE
+ GRAF2 DATASET USE
+
+ERPT237I END OF CANRPT EXPANSION
```


CICS.TRANGETM Report

The CICS.TRANGETM report graphs the average number of GETMAINS performed by each active transaction per use. This information is similar to the information shown by the online TEMPSTG command.

The following commands generate a CICS.TRANGETM report:

```
CANRPT CICS.TRANGETM
      FROM 01/06/03 08.00.00
      TO   01/06/03 09.00.00
END
RUN
```

The CICS.TRANGETM command expands as follows:

```
CICS.TRANGETM
ERPT237I CANNED REPORT EXPANSION
+ *****
+ ***
+ ** GRAF THE AVERAGE NUMBER OF **
+ ** GETMAINS DONE BY EACH **
+ ** TRANSACTION EACH TIME IT **
+ ** WAS USED. **
+ ***
+ *****
+ GRAF TRAN GETM
+
+ERPT237I END OF CANRPT EXPANSION
```


CICS.TRANLIFE.RANGE Report

The CICS.TRANLIFE.RANGE report shows transaction usage broken down into specific ranges of transaction lifetime.

The following commands generate a CICS.TRANLIFE.RANGE report:

```
CANRPT CICS.TRANLIFE.RANGE
  SHIFT 08:00:00 17:00:00
  FROM 12/06/03
  TO 12/09/03
  EACH HOUR
END
RUN
```

The CANRPT CICS.TRANLIFE.RANGE command expands as follows:

```
CANRPT CICS.TRANLIFE.RANGE
ERPT237I CANNED REPORT EXPANSION:
+*-----*
+* TRANSACTION USE BROKEN DOWN INTO *
+* SPECIFIC TRANSACTION LIFETIME GROUPS *
+*-----*
+TAB DATETIME TRAN USE AND,
+TRAN USE RATE HDR2(RATE) AND,
+TRAN USE RANGE(TLIFE1),
+ HDR1(TRAN USE) HDR2(0-1 SEC) AND,
+TRAN USE PCT RANGE(TLIFE1),
+ HDR1(% USE) HDR2(0-1 SEC) AND,
+TRAN USE RANGE(TLIFE3),
+ HDR1(TRAN USE) HDR2(1-3 SEC) AND,
+TRAN USE PCT RANGE(TLIFE3),
+ HDR1(% USE) HDR2(1-3 SEC) AND,
+TRAN USE RANGE(TLIFE5),
+ HDR1(TRAN USE) HDR2(3-5 SEC) AND,
+TRAN USE PCT RANGE(TLIFE5),
+ HDR1(% USE) HDR2(3-5 SEC) AND,
+TRAN USE RANGE(TLIFE10),
+ HDR1(TRAN USE) HDR2(5-10 SEC) AND,
+TRAN USE PCT RANGE(TLIFE10),
+ HDR1(% USE) HDR2(5-10 SEC) AND,
+TRAN USE RANGE(TLIFE+),
+ HDR1(TRAN USE) HDR2(10+ SEC) AND,
+TRAN USE PCT RANGE(TLIFE+),
+ HDR1(% USE) HDR2(10+ SEC)
+
+RANGE TRAN LIFE 0-1 ID TLIFE1
+RANGE TRAN LIFE 1-3 ID TLIFE3
+RANGE TRAN LIFE 3-5 ID TLIFE5
+RANGE TRAN LIFE 5-10 ID TLIFE10
+RANGE TRAN LIFE 10-999 ID TLIFE+
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICS.TRANLIFE.RANGE command generates a report like the following partial report:

SHIFT NONE		EACH HOUR		PERIOD NONE		TRAN USE		% USE		TRAN USE		% USE		TRAN USE		% USE	
DATE	TIME	USE	RATE	0-1 SEC	0-1 SEC	1-3 SEC	1-3 SEC	3-5 SEC	3-5 SEC	5-10 SEC	5-10 SEC	10+ SEC	10+ SEC	USE	RATE	0-1 SEC	0-1 SEC
12/06/03	08.00	26	0.01	9	34.6	0	0.0	0	0.0	1	3.8	16	61.5				
12/06/03	09.00	17	0.00	0	0.0	0	0.0	0	0.0	0	0.0	17	100.0				
12/06/03	10.00	16	0.00	0	0.0	0	0.0	0	0.0	0	0.0	16	100.0				
12/07/03	11.00	48	0.01	21	43.8	4	8.3	1	2.1	4	8.3	17	35.4				
12/07/03	12.00	105	0.03	71	67.6	6	5.7	0	0.0	2	1.9	26	24.8				
12/07/03	13.00	149	0.04	110	73.8	5	3.4	0	0.0	2	1.3	31	20.8				
12/07/03	14.00	151	0.04	104	68.9	7	4.6	1	0.7	1	0.7	38	25.2				
12/07/03	15.00	194	0.05	155	79.9	2	1.0	0	0.0	0	0.0	37	19.1				
12/07/03	16.00	228	0.06	188	82.5	1	0.4	2	0.9	1	0.4	36	15.8				
12/07/03	17.00	92	0.03	66	71.7	1	1.1	0	0.0	1	1.1	24	26.1				
12/08/03	08.00	119	0.03	88	73.9	3	2.5	2	1.7	0	0.0	26	21.8				
12/08/03	09.00	84	0.02	56	66.7	1	1.2	0	0.0	0	0.0	27	32.1				
12/08/03	10.00	56	0.02	39	69.6	0	0.0	2	3.6	0	0.0	14	25.0				
12/08/03	11.00	18	0.01	9	50.0	1	5.6	0	0.0	2	11.1	6	33.3				
12/08/03	12.00	63	0.02	40	63.5	1	1.6	0	0.0	1	1.6	21	33.3				
12/08/03	13.00	39	0.01	21	53.8	0	0.0	0	0.0	1	2.6	17	43.6				
12/09/03	14.00	120	0.03	103	85.8	0	0.0	0	0.0	0	0.0	17	14.2				
12/08/03	15.00	20	0.01	4	20.0	0	0.0	0	0.0	0	0.0	16	80.0				
12/08/03	16.00	20	0.01	4	20.0	0	0.0	0	0.0	0	0.0	16	80.0				
12/08/03	17.00	22	0.01	6	27.3	0	0.0	0	0.0	0	0.0	16	72.7				
12/09/03	08.00	22	0.01	7	31.8	1	4.5	0	0.0	0	0.0	14	63.6				
12/09/03	09.00	22	0.01	6	27.3	0	0.0	0	0.0	0	0.0	16	72.7				
12/09/03	10.00	22	0.01	4	18.2	1	4.5	1	4.5	0	0.0	16	72.7				
12/09/03	11.00	22	0.01	7	31.8	0	0.0	0	0.0	0	0.0	14	63.6				
12/09/03	12.00	20	0.01	4	20.0	0	0.0	0	0.0	0	0.0	16	80.0				
12/09/03	13.00	20	0.01	4	20.0	0	0.0	0	0.0	0	0.0	16	80.0				
12/09/03	14.00	88	0.02	48	54.5	4	4.5	3	3.4	3	3.4	28	31.8				
12/09/03	15.00	342	0.10	244	71.3	34	9.9	7	2.0	10	2.9	47	13.7				
12/09/03	16.00	2417	0.67	2118	87.6	114	4.7	30	1.2	29	1.2	126	5.2				
12/09/03	17.00	4122	1.26	3445	83.6	269	6.5	98	2.4	94	2.3	216	5.2				
PERIOD		8684	0.08	6981	80.4	455	5.2	147	1.7	152	1.8	943	10.9				

CICS.TRANPROG Report

The CICS.TRANPROG report displays information about all active transactions and their associated programs, sorted first by transaction use and then by program use. The transactions with the highest activity are listed first.

The following commands generate a CICS.TRANPROG report:

```
CANRPT CICS.TRANPROG
END
RUN
```

The CANRPT CICS.TRANPROG command expands as follows:

```
CANRPT CICS.TRANPROG
ERPT237I CANNED REPORT EXPANSION
+*-----*
+* DISPLAY IN TABULAR FORMAT ALL ACTIVE *
+* TRANSACTION AND ASSOCIATED PROGRAMS *
+*-----*
+TAB  TRANIDS  TRAN USE    AND,
+          PROG USE     AND,
+          PROG LINKS   AND,
+          PROG LOADS   AND,
+          PROG XCTLS   AND,
+          PROG FETCHS
+TAB2  PROGIDS  PROG USE    AND,
+          PROG LINKS   AND,
+          PROG LOADS   AND,
+          PROG XCTLS   AND,
+          PROG FETCHS  AND,
+          PROG RESP
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICS.TRANPROG command generates a report like the following partial report:

SHIFT	NONE												
EACH	DAY												
PERIOD	NONE												
TRAN ID	TRAN USE	PROG USE	PROG LINK	PROG LOAD	PROG XCTL	PROG FETCH	PROG ID	PROG USE	PROG LINK	PROG LOAD	PROG XCTL	PROG FETCH	PROG RESP
FXFR	2482	2484	0	2	2482	3	FXFRPP	2482	0	0	2482	1	0.315
							FXOPTAB	1	0	1	0	1	0.000
							FXGSRP	1	0	1	0	1	0.000
							SUMMARY	2484	0	2	2482	3	0.314
PSPI	939	5026	29	4058	939	0	C00M001M	3119	0	3119	0	0	0.000
							C00M001	939	0	939	0	0	0.000
							C02001	939	0	0	939	0	0.210
							REPDATA	29	29	0	0	0	0.110
SUMMARY	5026	29	4058	939	0	0.040							
CSNE	699	2075	720	656	699	0	DFHZNEP	720	720	0	0	0	0.002
							DFHZNAC	699	0	0	699	0	0.014
							VGBNET	656	0	656	0	0	0.000
SUMMARY	2075	720	656	699	0	0.005							
POSS	610	2544	323	1611	610	2	C00H000M	862	0	862	0	0	0.000
							C00000	610	0	0	610	0	0.095
							C00H000	565	0	565	0	0	0.000
							REPDATA	323	323	0	0	0	0.119
							C00M000	92	0	92	0	2	0.000
							C00M000M	92	0	92	0	0	0.000
SUMMARY	2544	323	1611	610	2	0.038							
X25T	546	8193	7049	0	1144	10	VGACTL	2597	2597	0	0	0	0.007
							VGATEWAY	1509	1509	0	0	0	0.019
							REPDATA	961	961	0	0	0	0.094
							VGA0101	546	546	0	0	0	0.767
							VGBUD01	546	0	0	546	0	0.781
							VGBEGIN	542	542	0	0	0	0.002
							VGBCG08	539	539	0	0	0	0.008
							C13004	190	0	0	190	0	0.695
							C13002	138	0	0	138	0	0.036
							VGCHECK	136	136	0	0	0	0.069
							C13001	125	0	0	125	2	0.036
						
						
						
SUMMARY	8193	7049	0	1144	10	0.169							
SUMMARY	5276	20322	8121	6327	5874	15	63167	37298	15331	10890	338	2.618	

CICS.TRANRESP Report

The CICS.TRANRESP report graphs the average transaction lifetime for each transaction that was used. It includes a sub-graph for each transaction showing terminal response time for each terminal that used the transaction.

The CICS.TRANRESP report may be very long, and it may be difficult to determine the relative lifetimes of transactions if the report contains transactions with widely diverse results. To create a more focused report, use the commands in the report's expansion rather than the CANRPT command, and add operands to the GRAF and GRAF2 commands. The operands of these commands are explained in the chapter titled "Graph Reports."

The following commands generate a CICS.TRANRESP report:

```
CANRPT CICS.TRANRESP
      TO 12/31/03 23.59.59
END
RUN
```

The CANRPT CICS.TRANRESP command expands as follows:

```
CANRPT CICS.TRANRESP
ERPT237I CANNED REPORT EXPANSION:
+ *****
+ ***
+ ** BREAK DOWN TRANSACTIONS BY **
+ ** THEIR LIFETIME AND SHOW **
+ ** TERMINAL RESPONSETIME FOR **
+ ** EACH TERMINAL USED BY EACH **
+ ** TRANSACTION. **
+ ***
+ *****
+ GRAF TRAN LIFE
+ GRAF2 TERM RESP
+
+ERPT237I END OF CANRPT EXPANSION
```


CICS.TRANSTAT Report

The CICS.TRANSTAT report shows how each active transaction in your system performed with respect to seven variables, sorted by maximum response times, from highest to lowest.

The following commands generate a CICS.TRANSTAT report:

```
CANRPT CICS.TRANSTAT
  TO 12/31/03 23.59.59
END
RUN
```

The CANRPT CICS.TRANSTAT command expands as follows:

```
  CANRPT CICS.TRANSTAT
  ERPT237I CANNED REPORT EXPANSION
+*****
+***                                     ***
+** SHOW IN TABULAR FORMAT WHAT TRANSACTIONS WERE **
+** USED, HOW EACH TRANSACTION PERFORMED,          **
+** AND WHAT RESOURCES IT HAS USED.                **
+***                                     ***
+*****
+TABULATE TRANIDS,
+   TRAN USE                                     AND,
+   TRAN USE PCT HDR1(TRAN) HDR2(%USE)          AND,
+   TRAN LIFE   HDR1(TRAN) HDR2(AVE-RT)        AND,
+   TRAN MAXLIFE HDR1(TRAN) HDR2(MAX-RT)       AND,
+   TRAN WAITS                                     AND,
+   TRAN STOR   HDR1(TRAN) HDR2(STOR/BYTES)    AND,
+
+   TERM USE    HDR1(TERM) HDR2(USE)           AND,
+   TERM MESSAGEs,
+   TERM RESP   HDR1(TERM) HDR2(RT-AVE)        AND,
+   TERM MAXRESP HDR1(TERM) HDR2(RT-MAX)       AND,
+
+   DATASET USE HDR1(FILE) HDR2(USAGE)        AND,
+   DATASET SERVTIME
+
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICS.TRANSTAT command generates a report like the following partial report:

SHIFT EACH PERIOD	NONE DAY NONE	TRAN ID	TRAN USE	TRAN %USE	TRAN AVE-RT	TRAN MAX-RT	TRAN WAITS	TRAN STOR/BYTES	TERM USE	R4READS	TERM RT-AVE	TERM RT-MAX	FILE USAGE	FILE I/O TIME
KCP		738	39.4	304.1	369.4			1940						
TCP		737	39.4	304.1	369.3	306		420						
IESX		58	3.1	0.732	3.284	4		3422						
CSNE		41	2.2	0.026	0.353	1		2176						
EXPC		33	1.8	0.179	2.072	1		3372	33	66	0.179	2.072		
CSPQ		30	1.6	0.038	0.461			216						
I\$\$P		28	1.5	0.015	0.194	6		3506			0.194			
IESW		27	1.4	0.111	0.976	30		70097	26	48	0.116	0.976	396	0.003
IE\$1		27	1.4	0.012	0.146			13088	1	1	0.321	0.146	4	0.002
CSSN		25	1.3	0.243	1.874	15		52983	31	43	0.094	0.563	256	0.003
ALXP		25	1.3	0.031	0.154	19		8887	25	50	0.031	0.154	463	0.001
IEGM		16	0.9	0.262	0.995	1		1876	14	14	0.274	0.992		
IESI		16	0.9	0.022	0.180	1		10704	12	28	0.029	0.180	10	0.003
IES6		13	0.7	0.018	0.058	2		16216	11	24	0.022	0.058	13	0.005
CATD		10	0.5	0.175	0.548	5		14458						
SCTY		10	0.5	0.171	0.702	6		17375	10	20	0.170	0.702	28	0.002
aLxp		8	0.4	0.138	0.879	2		8661	8	16	0.138	0.879	5	0.006
CRSQ		6	0.3	0.003	0.007			80						
I\$\$5		4	0.2	3.421	6.626	1		136						
I\$\$3		4	0.2	0.004	0.005			3168	1	1	0.016	0.005		
CSTA		3	0.2	0.305	0.556	1		176						
PF3		3	0.2	0.026	0.050	2		13427	3	6	0.026	0.050	3	0.015
XCOM		2	0.1	0.105	0.183	1		12412	2	4	0.105	0.183		
I\$\$1		2	0.1	0.003	0.004			3768		2		0.004		
CSGM		2	0.1	0.003	0.004			1284			0.004			
S140		1	0.1	58305	58305	9		597k					8	
I\$\$Q		1	0.1	265.9	265.9	1								
IE\$2		1	0.1	0.006	0.006			7200	1	1	0.006	0.006		
SUMMARY		1871	100.0	271.1	58305	122		4033	178	324	0.117	2.072	1186	0.002

CICS.TRANTERM Report

The CICS.TRANTERM report displays information about all active transactions and their associated terminals, sorted first by transaction use and then by terminal use. The transactions with the highest activity are listed first.

The following commands generate a CICS.TRANTERM report:

```
CANRPT CICS.TRANTERM
END
RUN
```

The CANRPT CICS.TRANTERM command expands as follows:

```
CANRPT CICS.TRANTERM
ERPT237I CANNED REPORT EXPANSION
+*-----*
+* DISPLAY IN TABULAR FORMAT ALL ACTIVE *
+* TRANSACTION AND ASSOCIATED TERMINALS *
+*-----*
+TAB  TRANIDS  TRAN USE          AND,
+          TRAN LIFE           AND,
+          TRAN MAXLIFE        AND,
+          TERM RESP           AND,
+          TERM MAXRESP        AND,
+          TERM IO
+TAB2  TERMIDS  TRAN USE          AND,
+          TRAN LIFE           AND,
+          TRAN MAXLIFE        AND,
+          TERM RESP           AND,
+          TERM MAXRESP        AND,
+          TERM IO
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT CICS.TRANTERM command generates a report like the following partial report:

SHIFT	NONE												
EACH	DAY												
PERIOD	NONE												
TRAN ID	TRAN USE	TRAN LIFE	TRAN MAX LIFE	TERM RESP	TERM MAX RESP	TERM IO	TERM ID	TRAN USE	TRAN LIFE	TRAN MAX LIFE	TERM RESP	TERM MAX RESP	TERM IO
FXFR	2482	0.361	38.82	0.361	38.82	4964	F014	2482	0.361	38.82	0.361	38.82	4964
PSPI	939	0.248	3.046	0.068	2.954	3611	8A21	197	0.219	1.514	0.066	1.463	715
							F006	116	0.327	3.046	0.067	2.954	561
							0A09	67	0.198	1.723	0.054	1.677	254
							0A02	64	0.224	0.814	0.058	0.756	261
							8A03	60	0.228	1.024	0.053	0.820	256
							8A09	57	0.163	0.516	0.048	0.456	190
							F012	41	0.271	1.946	0.046	1.802	198
							SUMMARY	602	1115	0.000	879.8	0.000	2435
CSNE	699	0.026	1.720	0.000	0.000	0	N/A	699	0.026	1.720	0.000	0.000	0
POSS	610	0.120	2.387	0.103	2.375	1037	8A21	78	0.092	0.310	0.078	0.257	123
							4A01	67	0.127	0.490	0.116	0.481	102
							0A09	62	0.088	0.593	0.071	0.593	109
							0A19	44	0.066	0.154	0.059	0.143	72
							8A09	42	0.122	0.764	0.106	0.732	68
							0A02	35	0.169	2.387	0.146	2.375	60
							8A26	32	0.099	0.270	0.088	0.247	52
							SUMMARY	360	299.5	0.000	271.0	0.000	586
X25T	546	0.818	14.43	0.765	14.43	1132	X253	389	0.901	14.43	0.849	14.43	804
							X254	138	0.635	5.912	0.577	5.912	290
							X255	19	0.434	1.102	0.434	1.102	38
							SUMMARY	546	0.818	14.43	0.765	14.43	1132
SUMMARY	5276	0.316	0.000	0.245	0.000	10744		8684	87.43	46487	0.275	56.39	20183

CICS.TRANTIO Report

The CICS.TRANTIO report graphs the average transaction usage for each transaction, together with a sub-graph of terminal I/O for each transaction.

The following commands generate the CICS.TRANTIO report:

```
CANRPT CICS.TRANTIO
  FROM 01/06/03 09.00.00
  TO   01/06/03 09.30.00
END
RUN
```

CICS.TRANTIO command expands as follows:

```
CANRPT CICS.TRANTIO
ERPT237I CANNED REPORT EXPANSION:
+ *****
+ ***
+ ** GRAF TERMINAL IOS FOR EACH **
+ ** TERMINAL A TRANSACTION USED **
+ ** AND GRAF TRANSACTION USE **
+ ** FOR EACH TRANSACTION USED. **
+ ***
+ *****
+ GRAF TRAN USE
+ GRAF2 TERM IO
+
+ERPT237I END OF CANRPT EXPANSION
```


CICS.TRANWAIT Report

The CICS.TRANWAIT report graphs the average number of times that each active transaction waited for any type of resource.

The following commands generate a CICS.TRANWAIT report:

```
CANRPT CICS.TRANWAIT
  FROM 01/06/03 11.00.00
  TO   01/06/03 11.30.00
END
RUN
```

The CANRPT CICS.TRANWAIT command expands as follows:

```
CANRPT CICS.TRANWAIT
ERPT237I CANNED REPORT EXPANSION:
+ *****
+ ***
+ ** SHOW GRAPHICALLY THE NUMBER **
+ ** OF WAITS EACH TRANSACTION **
+ ** HAD.
+ ***
+ *****
+ GRAF TRAN WAITS
+
+ERPT237I END OF CANRPT EXPANSION
```


VSE.CICS.STAT Report

The VSE.CICS.STAT report contains system information on both VSE and CICS.

The following commands generate a VSE.CICS.STAT report. To run this report, you must include an INPUT command specifying both the Unicenter CA-Explore for VSE flashback file and the Unicenter CA-Explore for CICS flashback file.

```
INPUT(EVSEFBK,EXPCFBK)
CANRPT VSE.CICS.STAT
END
RUN
```

The CANRPT VSE.CICS.STAT command expands as follows:

```
CANRPT VSE.CICS.STAT
ERPT237I CANNED REPORT EXPANSION
*-----*
* DISPLAY BOTH VSE SYSTEM RESOURCES *
* AND CICS RESOURCES *
*-----*
+TITLE2 VSE SYSTEM AND CICS STATISTICS
+TAB DATETIME VSE CPU ACTIVE AND,
+ VSE CPU UTILIZATION AND,
+ VSE PAGE FAULTS RATE,
+ HDR1(P-FAULT) HDR2(RATE) AND,
+ VSE DEVICE SIO,
+ HDR1(TOTAL) HDR2(IO) AND,
+ VSE DISK SIO HDR1(DISK) HDR2(IO) AND,
+ VSE DISK SIO RATE,
+ HDR1(DISK-IO) HDR2(RATE) AND,
+ VSE DISK SERVTIME,
+ HDR1(DISK) HDR2(SERVTIME) AND,
+ TRANSACTION USE AND,
+ TRANSACTION USE RATE HDR2(RATE) AND,
+ TERMINAL RESPONSE AND,
+ FILE REQUESTS AND,
+ FILE SERVTIME
+ERPT237I END OF CANRPT EXPANSION
```

The CANRPT VSE.CICS.STAT command generates a report like the following partial report:

DATE	TIME	CPU ACTIVE	CPU %ACTIVE	P-FAULT RATE	TOTAL IO	DISK IO	DISK-IO RATE	DISK SERVTIME	TRAN USE	TRAN RATE	TERM RESPTIME	FILE USE	REQUEST SERVTIME
01/17/03	00.00	01:08:39	10.6	0.09	50,465	41,186	1.03	0.027	457	0.01	1.421	22197	0.005
01/18/03	00.00	01:30:31	12.0	0.08	135,482	119,159	1.38	0.022	891	0.01	1.289	57343	0.008
01/19/03	00.00	01:02:43	10.8	0.06	98,341	70,465	0.98	0.007	521	0.01	1.365	26901	0.009
01/20/03	00.00	00:17:05	3.1	0.02	4,989	4,034	0.45	0.021	68	0.00	0.642	1900	0.003
01/21/03	00.00	00:03:11	0.7	0.02	845	621	0.67	0.020	21	0.00	0.710	217	0.003
01/22/03	00.00	02:13:32	16.4	0.12	273,101	224,178	1.93	0.028	2018	0.02	1.193	97045	0.010
01/23/03	00.00	01:49:29	13.0	0.15	187,422	151,342	1.60	0.030	1359	0.02	1.544	72663	0.009
01/24/03	00.00	00:08:38	8.9	0.05	53,985	36,985	0.91	0.030	432	0.01	1.391	21988	0.007
PERIOD		08:10:48	11.2	0.08	804,630	647,970	1.21	0.024	5767	0.01	1.331	300K	0.009

Command Descriptions and Syntax

This chapter describes all report writer commands and explains the syntax of each.

Some commands require you to specify at least one variable. See the chapter titled “Variables” for a detailed description of report writer variables.

Some commands allow you to specify one or more ID options either with the command or with the variables specified with the command. For a complete list of ID options and an explanation of how to use them, see the chapter titled “ID Options.” The ID options that are valid for each class of variable are listed in the chapter titled “Variables.”

For further examples of the primary commands, including sample reports, see the following chapters:

“Tabular Reports”

“Plot Report”

“Graph Reports”

“Flashback Reports”

Understanding Syntax Rules

Unicenter CA-Explore for CICS history reporting commands must follow certain syntax rules. If the report writer encounters a syntax error, it displays a dollar sign (\$) underneath the unexpected word or character, along with a message about what may be wrong. The report writer continues to read your commands to look for more syntax errors, but it does not generate a report. Follow these command syntax rules:

- Commands and their operands can be entered in either uppercase or lowercase letters.
- Each command must go on a new line. Do not enter more than one command on one line.
- Commands must be the first (leftmost) word on a line. You can start a command in any column, but it must always be the first word on the line.
- If you run out of space on one line for a command and its operands, you can continue it onto the next line by typing a comma after the last (rightmost) word on the line. You cannot split words with a continuation comma or a hyphen. When you continue a command, you can start the continued part of the command anywhere, as long as it is the first (leftmost) word on the line.
- Blanks, commas, or parentheses to be considered different words must separate operands and commands. Use one or more blanks to separate all words not separated by commas or parentheses. If you must use commas or parentheses, you do not have to separate words with blanks. Multiple blank spaces are treated as single blank spaces. You can type as many blanks as you want between words.
- Blank lines are ignored. You can include blank lines anywhere, in any number, even between a command and its continuation.
- Control statements are processed in the order in which they are read. Reports are printed in the order you request them.
- Any line whose first (leftmost) nonblank character is an asterisk (*) is a comment. The asterisk may be in any column. Anything after the asterisk is ignored.
- You can embed comments anywhere except between a command and its continuation lines. You cannot continue a comment with a continuation comma. To continue a comment, repeat the asterisk for each line of the comment.

Applying Syntax Conventions

In this manual, command syntax is shown using the conventions in the following table:

Convention	Explanation	Sample Syntax	Sample Entry
Uppercase	Uppercase letters indicate you must enter the entire word.	PLOT	PLOT
Lowercase	Variables are presented in lowercase letters and are italicized. You must substitute specific information for each variable.	PLOT <i>variable idoption(identifier)</i>	PLOT TRAN USE TRANID(CINQ)
Mixed case	Mixed case indicates you can enter either the uppercase portion or the entire word.	PerCenT	PCT
{ }	Braces enclose options that you are required to enter. Enter exactly one of the items listed vertically. Enter one or more of the items listed horizontally.	SCALE({ <i>mX</i> }, { <i>n</i> }, {COMMON})	SCALE(3X, COMMON)
[]	Brackets enclose options that you are not required to enter. You can enter one or none of the options listed vertically. You can enter more than one of the options listed horizontally, in the order indicated.	[HDR1(<i>text</i>)] HDR2(<i>text</i>) <i>n</i>	[FOR <i>n</i>] [TOP <i>n</i>] [BOTTOM] HDR2(Rate) TOP 5
()	Parentheses indicate that you must enter parentheses around the item as shown.	[SCALE({ <i>mX</i> }, { <i>n</i> }, {COMMON})]	SCALE(3X)
	An or-sign indicates a choice between related items. Enter only one of the items.	[PGMSTOR=[NO YES]]	PGMSTOR=NO
—	Underscoring indicates the default. If you want a command's underscored parameters only, you do not have to specify the command.	INPUT {(TAPEIN) } {(EXPCFBK)}	No command needed to specify INPUT(TAPEIN)
...	An ellipsis indicates that the preceding item can be repeated.	TAB <i>variable</i> [AND, ...]	TAB TRAN USE AND, TRAN LIFE

Using Global Commands

Global commands set values that stay in effect for all reports printed in one run. For example, if you use the LINECOUNT command at the beginning of a run to tell the report writer how many lines are on one page, that number of lines applies to all reports in that run.

You can insert global commands anywhere among the report commands. With the exception of the GLOBAL command, global commands affect all reports in the run without regard to placement. The GLOBAL command affects only reports that come after it.

The following table briefly describes the history reporting global commands:

Command	Function
CARDLEN	Sets the number of characters in each line the report writer will read. This is useful when you want the report writer to ignore sequence numbers.
COUNT	Specifies the number of input records the report writer is to read. This command is useful for testing new reports.
DEFINE	Creates new ID options and variables.
GLOBAL	Makes a command apply to all subsequent reports. This command helps you avoid duplicating code when you create multiple reports.
GOTO	Causes the report writer to ignore all subsequent commands until the specified label is read.
IF	Creates new identifiers based on other identifiers, or resets existing identifiers.
INPUT	Defines the input files to be used for the report (see OUTPUT command).
LINECOUNT	Specifies the number of lines to be printed per page of the report.
OPTION	Specifies global report options, such as how dates, null identifiers, and zero values are to be represented, whether summary lines, storage statistics, or dumps are produced, and whether messages are printed.
OUTPUT	Sends report output to the disk or tape file you specify.
RUN	Starts the printing of reports.

Using Report Commands

Report commands apply to a single report. These commands can be divided into two groups: *primary* report commands, which define the type of report, and *secondary* report commands, which alter or enhance the report.

The following table briefly describes the primary report commands:

Command	Function
CANRPT	Produces a canned report.
FLASHBACK	Prints detailed information about each transaction executed.
GRAF	Produces a graph report that shows the value of a variable for each resource of a single type (transaction, terminal, user, and so forth).
GRAF2	Produces a second graph under each graph line created by the GRAF command.
HPLOT	Produces a distribution plot of one variable against another.
INVOICE	Produces a tabular invoice report using the cost factors you specify, including a separate invoice for each identifier included in the report and an optional summary for all identifiers.
MPLOT	Plots multiple variables across time.
PLOT	Plots a variable across time.
PLOT2	Creates a second plot below each plot line created by the PLOT command.
TAB	Produces a table of the values of the variables you specify.
TAB2	Produces a second table for each row of a table created by the TAB command.
VLOT	Plots a variable vertically across time.
VLOT2	Plots a second variable on top of a plot created by the VLOT command.

List of Secondary Report Commands

The following table briefly describes the secondary report commands:

Command	Function
COMMENT	Prints a comment at the bottom of each page of the report.
COST	Assigns cost values to resources for charge-back.
EACH	Defines a time from minutes to years in which report data is to be displayed or organized.
END	Indicates the end of the current report description.
EXCLUDE	Defines a time range to be excluded from the report.
EXITRTN	Specifies the name of a user exit to be called after each record is read and before processing.
FROM	Defines the starting date and time of the report.
GROUP	Combines resources into performance groups.
HEADER	Assigns a header to appear above the column headings in a tabular report.
INCLUDE	Defines a time range to be included in the report.
PERIOD	Produces a summary of the plotted resource's average activity.
PRTEXT	Specifies the name of a user exit to be called before each line is printed.
RANGE	Defines a value range for any variable.
SELECT	Restricts a report to the specified identifiers.
SHIFT	Limits the report to a part of each day, typically a production shift.
TITLE1 TITLE2 TITLE3	Replaces the default title lines of a report with up to three lines of text you specify.
TO	Defines the ending date and time of the report.

Command Descriptions

The following sections describe command usage, syntax, and operands.

CANRPT Command

Only a brief description of the CANRPT command is given here. A more detailed description can be found in the chapter titled “Canned Reports.”

The CANRPT command generates a canned report.

Command Syntax and Operand

CANRPT *reportname*

Operand	Meaning
<i>reportname</i>	The name of the canned report you want to create

CARDLEN Command

The CARDLEN command sets the number of characters in each line the report writer will read. This is useful when you want the report writer to ignore sequence numbers.

The CARDLEN command is a global command. Entering it once sets the line length for all reports in the run. If you use this command, it should be the first command in your command stream; otherwise, the lines before it will be treated as if they are 80 characters long.

Command Syntax and Operand

CARDLEN *nn*

Operand	Meaning	Default
<i>nn</i>	The maximum number of columns (characters) per line the report writer is to read. <i>nn</i> must be an integer less than or equal to 80.	80

The following command specifies that the report writer is to read only 71 characters per line:

CARDLEN 71

COMMENT Command

The COMMENT command creates a comment line at the bottom of each page of the report.

Command Syntax and Operand

COMMENT *commentline*

Operand	Meaning
<i>commentline</i>	The comment that you assign to the report. The comment can be up to 60 characters long. Every character in <i>commentline</i> must be on the same line as the COMMENT command.

COST Command

The COST command makes charge-back statistics available for any resource.

Command Syntax and Operands

```
COST [variable cost-factor [idoption(identifiers)] [RANGE(min,max)]
[SELECT(name)]
    [variable cost-factor [DEFERRED]]
    [DISTRIBUTion value]
```

Operand	Meaning
<i>variable</i>	Any valid variable keyword. You can use variables defined with the DEFINE command if you specify the DEFERRED operand.
<i>cost-factor</i>	The monetary value associated with this resource. This value can include up to five decimal positions.
<i>idoption(identifiers)</i>	Any valid ID option, specified with any valid identifiers.
RANGE	Allows multiple cost factors for the same resource based on specified value ranges. (<i>min,max</i>) defines the value range. Specify a separate COST command for each range.
SELECT(<i>name</i>)	Allows you to apply a time specified with an INCLUDE or EXCLUDE command. <i>name</i> is the name assigned by the ID operand of the INCLUDE or EXCLUDE command. This operand functions similar to the SELECT option.
DEFERRED	By default, the cost-factor is applied to the data immediately, as each record is read. Specifying DEFERRED causes the report writer to apply the cost factor after all the data has been read. If you specify DEFERRED, you cannot specify the <i>idoption</i> , RANGE, or SELECT operands.
DISTRIBUTion <i>value</i>	Use to create a distributed cost report. <i>value</i> defines the cost you want the report rows to total. Specify the REPORT COST DISTRIBUTion variable to include a column in your report that shows the adjusted total cost for each row, with all rows adding up to the amount defined by <i>value</i> .

If you use the COST command with a variable you created using the DEFINE command, you must specify the DEFERRED operand.

Related Variables

Use the variable REPORT COST TOTAL to create a column showing **total** cost for each row in your report.

Use the variable REPORT COST DISTRIBUTion to create a column showing **proportional total** cost for each row in your report. Use this variable only if you specified the DISTRIBUTion operand.

Related Option

Use the COST option with a variable to show its cost value, computed using the cost factor you specify with the COST command. For example, you might specify a cost factor for the variable TRAN USE using the COST command. Then you could specify the variable and option TRAN USE COST with the TAB command to create a column in the report showing the cost for transactions for each identifier or time in the report.

Related Command

Use the command OPTION MONETARY=*x* to specify a one-character symbol for the monetary unit. The default setting is the dollar sign (\$).

Example

The COST commands in this example define the following cost factors: 1 cent per transaction, 25 cents per CPU second consumed, 0.1 cent for each file request, and 0.5 cent for each file add. The TAB command makes use of the four cost factors, creating a tabular report with a column showing the value of each variable for each transaction. The HDR1 and HDR2 operands assign those columns appropriate headings. The last variable, REPORT COST TOTAL, creates a column that shows the sum of the costs for each row in the report

```

COST TRAN USE .01
COST TRAN CPU .25
COST FILE USE .001
COST FILE ADDS .005

TAB TRANID TRAN USE
      TRAN USE COST HDR1(TRAN USE) HDR2(COST) AND,
      TRAN CPU AND,
      TRAN CPU COST HDR1(TRAN CPU) HDR2(COST) AND,
      FILE USE AND,
      FILE USE COST HDR1(FILE USE) HDR2(COST) AND,
      FILE ADDS AND,
      FILE ADDS COST HDR1(FILE ADDS) HDR2(COST) AND,
      REPORT COST TOTAL

END
RUN
    
```

The preceding commands create a report like the following:

SHIFT	NONE									
EACH	DAY									
PERIOD	NONE									
TRAN ID	TRAN USE	TRAN USE COST	TRAN CPU	TRAN CPU COST	FILE USE	FILE USE COST	FILE ADDS	FILE ADDS COST	TOTAL COST	
X25T	57	\$0.57	0.132	\$1.88	198	\$0.20	9	\$0.05	\$2.70	
CSNE	49	\$0.49	0.010	\$0.12	0	\$0.00	0	\$0.00	\$0.61	
TCP	46	\$0.46	0.572	\$6.57	241	\$0.24	0	\$0.00	\$7.27	
KCP	45	\$0.45	7.581	\$85.28	0	\$0.00	0	\$0.00	\$85.73	
PSPI	43	\$0.43	0.097	\$1.04	0	\$0.00	0	\$0.00	\$1.47	
POSS	37	\$0.37	0.035	\$0.32	21	\$0.02	0	\$0.00	\$0.71	
VGBC	13	\$0.13	0.055	\$0.18	0	\$0.00	0	\$0.00	\$0.31	
JJJ	10	\$0.10	0.221	\$0.55	0	\$0.00	0	\$0.00	\$0.65	
CSPQ	10	\$0.10	0.113	\$0.28	0	\$0.00	0	\$0.00	\$0.38	
CATD	9	\$0.09	0.105	\$0.24	0	\$0.00	0	\$0.00	\$0.33	
CSSN	8	\$0.08	0.071	\$0.14	0	\$0.00	0	\$0.00	\$0.22	
CSGM	8	\$0.08	0.007	\$0.01	0	\$0.00	0	\$0.00	\$0.09	
RAPL	4	\$0.04	0.011	\$0.01	0	\$0.00	0	\$0.00	\$0.05	
SPAM	3	\$0.03	0.132	\$0.10	3	\$0.00	0	\$0.00	\$0.13	
PINV	2	\$0.02	1.486	\$0.74	26	\$0.03	0	\$0.00	\$0.79	
PADI	2	\$0.02	0.350	\$0.17	0	\$0.00	0	\$0.00	\$0.19	
RAPP	2	\$0.02	0.098	\$0.05	0	\$0.00	0	\$0.00	\$0.07	
SUMMARY	348	\$3.48	1.123	\$195.41	489	\$0.49	9	\$0.05	\$199.43	

COUNT Command

COUNT command defines a maximum number of input records to be read for a report. After the specified number of records is read, an end-of-file condition is forced.

The COUNT command is a global command. Entering it once sets the maximum number of input records to be read for each report in the run. You can use this command to experiment with new reports that might generate more output than you want.

If a FROM command is specified, the counting of records begins with the first record meeting the earliest time specified in the FROM command. Otherwise, the counting of records begins with the first record read.

Command Syntax and Operands

COUNT *nnnnn* [SELECTED]

Operand	Meaning
<i>nnnnn</i>	The maximum number of records to be read
SELECTED	Specifies that only records that pass at least one selection criterion are to be counted when determining if <i>nnnnnn</i> records have been read

The following command specifies that the report writer is to stop reading records after reading fifty records that meet any selection criterion:

```
COUNT 50 SELECTED
```

DEFINE Command

Use the DEFINE command to create new ID options and variables for use in reports created with the TAB, TAB2, HPLOT, MPLOT, VPLOT, and VPLOT2 commands.

The DEFINE command is a global command. Any variables or ID options you define will be in effect for all reports in the run.

WARNING! When you use the DEFINE command to create variables, you may need to specify the FORMAT variable option with the variable in your report command. If you do not specify the FORMAT option, variable values will not include a decimal point.

Formatting Identifiers

The FORMAT option can be used with ID options to control the number of characters printed for each identifier.

Defining ID Options Using Existing ID Options

Use the following syntax to define new ID options using existing ID options:

```
DEFINE ID newid [oldid [(start, length)]] ['string'] ...
```

Operand	Meaning
ID	Requests the definition of a new ID option.
<i>newid</i>	The name of the new ID option you want to define.
<i>oldid</i>	The name of an existing ID option you want to be a component of the new ID option.
<i>start,length</i>	If you do not want to include the entire existing ID option in the new ID option, specify the starting position (<i>start</i>) and length (<i>length</i>) of the part of it you want to include.
<i>string</i>	A character string that you want to be a component of the new ID option. You must enclose the string in single quotes. The string can be more than one character in length, and it can include or consist entirely of blanks.

When defining a new ID option, you can include multiple character strings or existing ID options, in any order. When printed, components are concatenated without intervening spaces.

Defining Variables Using Existing Variables

Use the following syntax to define new variables using existing variables:

```
DEFINE VAR newvar = expression
```

Operand	Meaning
VAR	Requests the definition of a new variable.
<i>newvar</i>	The name of the new variable you want to define.
<i>expression</i>	<p>Any arithmetic expression, the result of which is the value of the new variable. You can use the following to create the expression:</p> <p>Existing variables</p> <p>The MAX, MIN, TOTAL, and RANGE variable options</p> <p>Valid ID options</p> <p>Constants</p> <p>You must delimit all variables, operands, and parentheses with blanks. You can use the following operators in the expression:</p> <ul style="list-style-type: none">+ Addition- Subtraction* Multiplication/ Division! Exponentiation() Parentheses

Defining ID Options or Variables Without Using Existing ID Options or Variables

Use the following syntax to define new ID options or variables without using existing ID options or variables. This allows you to report on any data field located in any record. The syntax is shown on several lines with a continuation comma at the end of each line except the last.

```
DEFINE XID|XVAR name PRODUCT=prod RECORD=rectype [SUBTYPE=subtype] OFFSET=data-offset LENGTH=data-length,
[TRIPLET=triplet-offset] [TYPE=data-type] [HDR1=header1] [HDR2=header2],
[OFFSET2=data-offset2] [LENGTH2=data-length2] [TRIPLET2=triplet-offset2],
[VALUE=expression] [FORMAT(output-format)] [CLASS=class]
```

Operand	Meaning
XID	Requests the definition of a new ID option.
XVAR	Requests the definition of a new variable.
<i>name</i>	The name of the new ID option or variable you are creating. For DEFINE XID commands, <i>name</i> must be a single word.
<i>prod</i>	Identifies the Unicenter CA-Explore for CICS product for which an ID option or variable is being defined. You can omit this operand if you use the PRODUCT operand of the OPTION command.
<i>rectype</i>	The record type. To specify a value in hexadecimal, prefix the value with the letter x, as in RECORD=x3C. You can omit this operand if you use the RECORD operand of the OPTION command.
<i>subtype</i>	The record subtype.
<i>data-offset</i>	The offset to the data field from the beginning of the record. To specify a value in hexadecimal, prefix the value with the letter x, as in OFFSET=x3C.
<i>data-length</i>	The length of the data field. If you specified XVAR, the maximum length is 4.
<i>triplet-offset</i>	The offset from the beginning of the record to the triplet information. To specify a value in hexadecimal, prefix the value with the letter x, as in TRIPLET=x3C.

Operand	Meaning
<i>data-type</i>	<p>For DEFINE XID commands, one of the following data types:</p> <p>BINARY – Binary</p> <p>CHARACTER – Character; this is the default for DEFINE XID commands</p> <p>HEX – Hexadecimal</p> <p>For DEFINE XVAR commands, one of the following data types:</p> <p>BINARY – Binary; this is the default for DEFINE XVAR commands</p> <p>CHARACTER – Character</p> <p>DECIMAL – Decimal numbers in character format</p> <p>HEX – Hexadecimal</p> <p>STCK – Converts a store-clock time-of-day (TOD) field into mm/dd/yy hh:mm:ss format</p> <p>STCK-DATE – Converts a store-clock TOD field into mm/dd/yy format.</p> <p>STCK-TIME – Converts a store-clock TOD field into hh:mm:ss format</p> <p>STCK-TIMEM – Converts a store-clock TOD field into hh:mm:ss:mmm format</p>
<i>header1</i>	A string of up to 8 characters to be used as the top column heading. The default is the null string.
<i>header2</i>	A string of up to 8 characters to be used as the bottom column heading. The default is the null string.
<i>data-offset2</i>	The offset to the data field from the beginning of the record. To specify a value in hexadecimal, prefix the value with the letter x, as in OFFSET=x3C.
<i>data-length2</i>	The length of the data field. If you specified XVAR, the maximum length is 4.
<i>triplet-offset2</i>	The offset from the beginning of the record to the triplet information. To specify a value in hexadecimal, prefix the value with the letter x, as in TRIPLET=x3C.

Operand	Meaning
<i>expression</i>	Any arithmetic expression using V1, V2, +, -, *, and /, where V1 represents the first data field (defined by OFFSET, LENGTH, TRIPLET) and V2 represents the second data field (defined by OFFSET2, LENGTH2, and TRIPLET2). The expression is processed left to right.
<i>output-format</i>	The default output pattern. Use the same syntax as the FORMAT variable option.
<i>class</i>	Specify one of the following for <i>class</i> : AVG – Average per record COUNT – Simple record count MAX – Save maximum value MIN – Save minimum value IMMEDIATE – Calculate the expression continuously during input record processing. This causes overhead and should not be used unless needed. One reason would be to allow a RANGE command to function with this defined variable.

The following examples show ways to enter the DEFINE command and the result of each:

Sample Entry	Result
DEFINE ID TRANID TRANID(1,4)	Redefines the length of the ID option TRANID from 8 characters to 4, starting with position 1.
DEFINE ID PROGID PROGID(3,4)	Redefines the ID option PROGID to use only 4 characters, starting with position 3.
DEFINE ID OPERID 'Operator ' OPERID	Redefines the ID option OPERID to include the word Operator and a blank before the operator ID.
DEFINE ID MYKEY TRANID ' ' TERMID	Creates a new ID option called MYKEY with components TRANID and TERMID and places a blank between the two components.
DEFINE VAR TRAN %CPUTIME = , (TRAN CPUTIME * 100) / TRAN LIFETIME	Creates the new variable TRAN %CPUTIME by multiplying TRAN CPUTIME by 100 and then dividing by TRAN LIFETIME. The continuation comma allows the command to be continued on a second line.
DEFINE VAR TRAN DCNTMAX = TRAN DCOUNT MAX	Creates the new variable DCNTMAX that yields the maximum number of transient datasets used by any one transaction.
DEFINE XID KEY19 PRODUCT=EXPC RECORD=19, OFFSET=X14 LENGTH=8 TYPE=HEX	Creates the new ID option KEY19, which is of record type 19, at offset X'14', of length 8, and of type hexadecimal. The continuation comma allows the command to be continued on a second line.
DEFINE XVAR TERM MYVAR, PRODUCT=EXPC RECORD=19 OFFSET=X28, LENGTH=2 TYPE=HEX HDR2=myvar	Creates the new variable TERM MYVAR. The continuation commas allow the command to be continued on subsequent lines.

EACH Command

The EACH command defines a time, from minutes to years, in which report data is to be organized. The effects of the EACH command depend on the report type, as follows:

- With tabular reports, the EACH command defines the time interval at which the table is to be repeated. You must allow for a DAY, DATE, or TIME column if you use the EACH command.
- For plot reports, use the EACH command to define the length of time represented by each line in the plot. You can plot by the hour, day, week, or month using the EACH command. The default is one day.

Graph reports break down activity by identifier during a given time period. If you do not specify an EACH command, the result is a single graph for the whole time covered in the report. Specifying an EACH command causes the report writer to produce a separate graph for each time specified by the EACH command.

You must have the appropriate amount of data in your input file for the EACH value you specify. For example, you cannot specify EACH YEAR if your input file has less than one year's data. If the input file does not have enough data, you may not receive a complete report.

WARNING! *If you use variables that use data collected at time intervals, such as INTerval variables, be aware that the report writer cannot report on a smaller time interval than the data in the input file. For example, if the input file contains data that was logged at half-hour intervals, you cannot specify an EACH value of less than 30 minutes. Generally, the EACH value you specify should be the same as, or a multiple of, your system data collection interval. This warning does not apply to PERFORMANCE record data because such data is not collected at specified time intervals.*

Specifying the First Day of a Week

By default, a week is Monday through Sunday. If you specify the EACH WEEK or EACH *nnn* WEEKs command and want weeks to begin on a different weekday, use the OPTION(WEEK=*weekday*) command to specify the first day of a week.

Command Syntax and Operands

EACH [*time-period* | RECORD] [AVERAGED] [SUMMARIZE]

Operand	Meaning
<i>time-period</i>	<p>One of the following base time units. <i>nnn</i> is the number of time intervals (up to 999) in the time unit you select. The default is DAY.</p> <ul style="list-style-type: none">■ MINUTE■ <i>nnn</i> MINUTEs■ HOUR■ <i>nnn</i> HOURs■ DAY■ <i>nnn</i> DAYs■ WEEK■ <i>nnn</i> WEEKs■ MONTH■ <i>nnn</i> MONTHs■ YEAR■ <i>nnn</i> YEARs
RECORD	<p>Defines the sample interval as each record, rather than as a time interval. This option is valid only for HPLOT reports; if you specify EACH RECORD for any other report, the command will be ignored. If you specify EACH RECORD for an HPLOT report, you must use the SCALE operand to specify a scale for the HPLOT variable.</p>

Operand	Meaning
AVERAGED	<p>Produces a report with one detail line for each time specified by the EACH command. The variable values in each line represent the average value of the variable for that time for all the input data.</p> <p>For example, if you create a tabular report and specify EACH 1 HOUR AVERAGED, the report will include 24 rows, one for each hour of the day. The first row would show the average value for each variable for the hour of midnight to 1 a.m. for all the days of input data.</p> <p>When you use the AVERAGED operand, the EACH value must be 24 hours or less.</p>
SUMMARIZE	<p>Similar to the AVERAGED operand, except that the data is totaled instead of averaged. Use this operand only with variables that represent counts (not averages).</p> <p>For example, if you specify EACH 1 HOUR SUMMARIZE for a tabular report, the report will include 24 rows, one for each hour of the day. The first row would show the total value for each variable for the hour of midnight to 1 a.m. for all the days of input data.</p>

To plot the value of a variable for each half-hour period, enter the following commands:

```
PLOT variable
    EACH 30 MINUTES
END
RUN
```

The following commands create a distribution plot using each record as the sample interval:

```
VPLOT variable1
HPLOT variable2 SCALE(nn)
    EACH RECORD
END
RUN
```

END Command

You must use the END command to end each report. Any commands specified after the END command are interpreted as applying to the next report. If you do not enter the END command between reports, the report writer issues a message that you are using multiple report commands for one report.

Command Syntax and Operand

END

The following is an example of the use of the END command:

```
TITLE1 REPORT1
PLOT variable1 AND variable2
    FROM 02/01/02 00.00.01
    TO   02/15/02 23.59.59
END

TITLE1 REPORT2
PLOT variable
    SHIFT 17:00:00 01:00:00
END
.
.
.
RUN
```

EXCLUDE Command

The EXCLUDE command defines time ranges to be excluded from a report.

You cannot use both INCLUDE and EXCLUDE commands in the same report. You can use multiple EXCLUDE commands to exclude multiple time ranges.

Command Syntax and Operands

```
EXCLUDE [day] [time - time] [ID name]
        [date]
        [date time - date time]
        [month]
```

Operand	Meaning
<i>day</i>	The day of the week (SUNday, MONday, TUESday, WEDnesday, THURsday, FRIday, or SATurday)
<i>time</i>	The time of day in <i>hh:mm:ss</i> format
<i>date</i>	The date in <i>mm/dd/yy</i> format
<i>ID name</i>	An identifying name to be associated with a specific setting of the EXCLUDE command, for use with the SELECT variable option
<i>month</i>	The month (JANuary, FEBruary, MARch, APRil, MAY, JUNE, JULy, AUGust, SEPTember, OCTober, NOVember, or DECember)

The following are examples of the EXCLUDE command:

```
EXCLUDE SATURDAY
```

```
EXCLUDE SUNDAY
```

```
EXCLUDE 07/04/00
```

```
EXCLUDE 12:00:00 - 13:00:00
```

```
EXCLUDE FRIDAY 16:00:00 - 17:00:00
```

```
EXCLUDE 08:00:00 - 15:59:59 ID 1STSHIFT
```

```
EXCLUDE 16:00:00 - 23:59:59 ID 2NDSHIFT
```

```
EXCLUDE JULY
```

EXITRTN Command

The EXITRTN command defines a user exit to be called after each record is read and before the report writer processes the record.

Supplied User Exit

One user exit, EXPGUX, is predefined for your use.

Command Syntax and Operand

EXITRTN(*userexit*)

Operand	Meaning
<i>userexit</i>	The name of the user exit

FLASHBACK DETAIL Command

Only a brief description of the FLASHBACK DETAIL command is given here.

The FLASHBACK DETAIL command generates detailed reports on transactions. It produces one page of output per transaction. The format of each page is fixed.

Transactions are presented in chronological order of ending time. You cannot sort flashback reports.

You can run flashback reports against the flashback file, the archive file, or a backup of the archive file (a log file).

WARNING! *Each flashback report must be run as a single report in a separate run. In other words, you cannot specify another primary report command (such as TAB or PLOT or another FLASHBACK DETAIL command) in a job stream with the FLASHBACK DETAIL command.*

Output Considerations

The FLASHBACK DETAIL command can generate enormous amounts of output if run against a very large input file. To avoid problems, use either the FROM and TO commands, the COUNT command, or the SELECT command to limit the output.

Command Syntax and Operand

```
FLASHBACK DETAIL [ idoption(identifiers) ... ]
```

Operand	Meaning
<i>idoption(identifiers)</i>	Any ID option followed by one or more identifiers. You can specify multiple ID options.

FLASHBACK LIST Command

Only a brief description of the FLASHBACK LIST command is given here.

The FLASHBACK LIST command generates detailed reports on transactions. It produces one line of output per transaction. You can change the number of columns per line by specifying variables with the FLASHBACK LIST command.

Transactions are listed in chronological order of ending time. You cannot sort flashback reports.

You can run flashback reports against the flashback file, the archive file, or a backup of the archive file (a log file).

WARNING! *Each flashback report must be run as a single report in a separate run. In other words, you cannot specify another primary report command (such as TAB or PLOT or another FLASHBACK LIST command) in a job stream with the FLASHBACK LIST command.*

Output Considerations

The FLASHBACK LIST command can generate enormous amounts of output if run against a very large input file. To avoid problems, use either the FROM and TO commands, the COUNT command, or the SELECT command to limit the output.

Command Syntax and Operands

```
FLASHBACK LIST [columns] [idooption[(identifiers)] [variable [options] [HDR1(text)] [HDR2(text)]
[AND, variable [options] [HDR1(text)] [HDR2(text)] [AND, ...]]
```

Operand	Meaning
<i>columns</i>	<p>One or more of the following, in any order, to indicate the first columns to appear in the report. If you do not specify any of these columns, the resulting report is organized by region ID and time.</p> <p>DATETIME—Creates columns for the date and time of the record.</p> <p>DATE—Creates a column for the date of the record.</p> <p>TIME—Creates a column for the time of the record.</p> <p>DAY—Creates a column for the day of the record (MONDAY, TUESDAY, and so forth).</p> <p>JOBNAME—Creates a column for the CICS jobnames.</p> <p>REGID—Creates a column for the CICS region IDs. Region IDs are set in the EXPCMIT with the REGID parameter.</p> <p>You can enter only one of DATETIME, DATE, TIME, and DAY. Because the width of the output page limits the number of columns that can appear in a report, you can use one of these operands only if you specify the variables to be included in your report and limit their number. The number of variables you can specify depends upon the width of the column for each.</p>
<i>idooption</i> (<i>identifiers</i>)	Lists all the resources (or just those specified by <i>identifiers</i>) for an ID option. You can specify multiple ID options.
<i>variable</i>	Any report writer variable. If you specify a variable, you must specify at least one ID option.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.
HDR1(<i>text</i>)	Overrides the default top heading for a column. Replace <i>text</i> with the new column heading. To remove the header, replace <i>text</i> with a null string, as in HDR1().
HDR2(<i>text</i>)	Overrides the default bottom heading for a column. Replace <i>text</i> with the new column heading. To remove the header, replace <i>text</i> with a null string, as in HDR2().
AND,	Signals that another variable will be specified. The comma after <i>AND</i> lets you continue to the next line.

FROM and TO Commands

The FROM and TO commands specify the period a report is to cover. The FROM command tells the report writer to exclude data less recent than the specified date and time. The TO command tells the report writer to exclude data more recent than the specified date and time. The TO date and time must be later than the FROM date and time.

The dates and times a report covers are printed at the top of each page, following the FROM and TO headings.

Command Syntax and Operand

```
FROM  time
TO    time
```

Operand	Meaning
<i>time</i>	The date and time that you specify, in either of two formats. You can specify an actual date and time , or you can specify a relative date and time

How FROM and TO Relate to the Log and Flashback Files

Reports cannot start earlier or end later than the data in the input file used. If the FROM and TO dates or times are outside the range of the data in the input file, the report will contain the data in the file closest to the dates and times specified.

Default Dates and Times

If no FROM and TO dates are specified, the report covers all records in the input file. If dates but no times are specified in the FROM and TO commands, the report covers all records within the specified dates.

Specifying an Actual Date and Time

The format for specifying an actual date and time is as follows:

mm/dd/yy hh:mm:ss

The operands for the actual date and time format are as follows:

Operand	Meaning
<i>mm/dd/yy</i>	The date in decimal digits, where <i>mm</i> is the month, <i>dd</i> is the day, and <i>yy</i> is the year. If a value is less than 10, you must include a leading zero (for example, specify 03/04/00 rather than 3/4/00). If your operating system uses the European date format or you specified it using the DATEFRMT operand of the OPTION command, you must enter dates in the format <i>dd/mm/yy</i> .
<i>hh:mm:ss</i>	The time in hours, minutes, and seconds. You can use periods instead of colons (<i>hh.mm.ss</i>). The hours are in military format, from 01 to 23 hours. The values for <i>mm</i> and <i>ss</i> cannot exceed 59. The default is 00:00:01.

Specifying a Relative Date and Time

If you specify actual dates and times in a report you plan to run frequently, you will have to change all of the dates in the job every time you run it. The alternative is to use the relative date and time format, which creates reports that start or end a certain number of minutes, hours, days, weeks, or months before the current date and time. The date and time you specify is always relative to the time that you run the report. Any INCLUDE and EXCLUDE commands preceding the report are honored.

The relative date and time format is as follows:

```
-n [MINUTES]
   [HOURS ]
   [DAYS ] [AT hh:mm:ss]
   [WEEKS ] [AT hh:mm:ss]
   [MONTHS ]
```

The operands for the relative date and time format are as follows:

Operand	Meaning
<i>-n</i>	The number of minutes, hours, days, weeks, or months to subtract from the time at which the report writer starts running. It must be an integer, prefixed by a minus sign (-). You can specify zero (-0).
MINUTES	Subtracts <i>n</i> minutes from the time at which the report writer starts running, rounded down to the nearest minute for FROM commands, and rounded up to the nearest minute for TO commands. If <i>-n</i> is zero (-0), the report writer rounds the current time down to the nearest minute.
HOURS	Subtracts <i>n</i> hours from the time at which the report writer starts running, rounded down to the nearest hour for FROM commands, and rounded up to the nearest hour for TO commands. If <i>-n</i> is zero (-0), the report writer rounds the current time down to the nearest hour.
DAYS	Subtracts <i>n</i> days from the current date and time, rounded down to the nearest day for FROM commands, and rounded up to the nearest day for TO commands. If <i>-n</i> is zero (-0), the report writer rounds the current date and time down to the start of the current day.
WEEKS	Subtracts <i>n</i> weeks from the current date and time, rounded down to the nearest week for FROM commands, and rounded up to the nearest week for TO commands. A week begins on Monday morning at 00:00:00. If <i>-n</i> is zero (-0), the report writer rounds the current date and time down to the start of the current week.
MONTHS	Subtracts <i>n</i> months from the current date and time, rounded down to the nearest month for FROM commands, and rounded up to the nearest month for TO commands. If <i>-n</i> is zero (-0), the report writer rounds the current date and time down to the start of the current month.
AT <i>hh:mm:ss</i>	Begins (or ends) the report at the time specified by <i>hh:mm:ss</i> on the day the report is run. The value is rounded down to the nearest day or week for both FROM and TO commands. Valid only with the DAYS and WEEKS operands.

Actual Dates and Times

To report from December 25, 2000, at 8:00 a.m. to January 1, 2001, use the following commands:

```
FROM 12/25/00 08:00:00
TO   01/01/01 00:00:00
```

Relative Dates and Times

To report from the start of the current week (Monday morning at 00:00:00), use the following command:

```
FROM -0 WEEKS
```

To report from yesterday at 00:00:00 to the current time, use the following commands. The report is rounded to the last hour.

```
FROM -1 DAYS
TO   -0 HOURS
```

To report from yesterday at 8:00 a.m. to today at 8:00 a.m., use the FROM and TO commands and the AT *hh:mm:ss* operand as follows:

```
FROM -1 DAYS AT 08:00:00
TO   -0 DAYS AT 08:00:00
```

To report from yesterday at 8:00 a.m. to yesterday at 4:00 p.m., use the following commands:

```
FROM -1 DAYS AT 08:00:00
TO   -1 DAYS AT 16:00:00
```

GLOBAL Command

The GLOBAL command assigns values to parameters of history reporting commands to be applied to all remaining reports in the run until another GLOBAL command resets the value. The value set by the GLOBAL command does **not** apply to subsequent reports in which the same history reporting command appears.

Command Syntax and Operand

GLOBAL *command*

Operand	Meaning
<i>command</i>	<p>One of the following commands, specified with any of its valid operands (for descriptions of command operands, see the specific section for that command in this chapter):</p> <ul style="list-style-type: none"> ■ COMMENT ■ COST ■ EACH ■ EXCLUDE ■ EXITRTN ■ FROM ■ GROUP ■ INCLUDE ■ PERIOD ■ PRTEXT ■ SHIFT ■ TITLE1 ■ TITLE2 ■ TITLE3 ■ TO

Temporarily Overriding a Global Value

To override a global value for one report, issue the command before the END command for that report. For example, if GLOBAL EACH 2 HOURS were set, you can override it within a report by specifying the report command EACH 4 HOURS with the commands for that report.

Removing a Global Value

You can remove a global value by issuing a GLOBAL command without operands. For example, specifying the command GLOBAL FROM removes any global value previously set for the FROM command.

GOTO Command

The GOTO command causes the report writer to ignore all subsequent commands until the specified label is read.

Command Syntax and Operand

```
GOTO .label
```

Operand	Meaning
<i>.label</i>	A period followed by a one- to eight-character label. You must also specify a period followed by this label on a separate line later in the job stream to indicate where the report writer should resume processing commands.

The following commands produce the canned reports CICS.OPERSTAT and CICS.FILEWAIT. The commands between GOTO .MYLABEL and .MYLABEL are ignored by the report writer, so the canned report CICS.INTERVAL.STAT is not produced and no syntax error is generated for the missing operand for the EACH command.

```
CANRPT CICS.OPERSTAT
END

GOTO .MYLABEL

CANRPT CICS.INTERVAL.STAT
  EACH
END

.MYLABEL

CANRPT CICS.FILEWAIT
END

RUN
```

GRAF Command

Only a brief description of the GRAF command is given here.

The GRAF command produces a single-line bar graph of the performance of each identifier associated with a variable. For example, using the GRAF command with a TERMINAL variable produces a bar graph of the variable for each terminal.

Command Syntax and Operands

```
GRAF [idoption[(identifiers)]] variable [options] [SCALE({n},{mX})] [SORT order] [FOR n] CHAR1(x)
```

Operand	Meaning
<i>idoption</i> (<i>identifiers</i>)	Any ID option that is valid for the variable, specified with or without identifiers. Specifying this operand causes the resulting report to be organized by the specified ID option and limited to the identifiers specified by <i>identifiers</i> . If you do not specify an ID option, all identifiers of the type associated with the variable are included in the report. For example, a graph of a TRANsaction variable by default graphs the value of that variable for each transaction. Specifying the ID option and identifier TRANID(TRAN1) graphs the value of the variable for transaction TRAN1 only; specifying the ID option TERMID graphs the value for each terminal.
<i>variable</i>	Any report writer variable.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.

Operand	Meaning
SCALE(<i>n</i>) or SCALE(<i>mX</i>) or SCALE(<i>n,mX</i>)	<p>By default, the highest value of a bar graph's axis is set to the highest value of the graphed variables. For variables that represent percentages, the default maximum value is 100%.</p> <p>To override this dynamic scale, specify <i>n</i> as the maximum value. If the variable that you choose produces integers, <i>n</i> must be an integer. You cannot use more than three digits to the right of the decimal point when specifying <i>n</i>.</p> <p>In some cases, bar graphs in a graph report will be too short to appear. To display these graphs, you can cause each plot character to appear as <i>m</i> characters by specifying the multiplier <i>mX</i>, where <i>m</i> is an integer. Since the resulting bar graphs will no longer correspond numerically to the scale, this is useful only for determining relative performance.</p>
SORT <i>order</i>	<p>Sorts the report. Replace <i>order</i> with one of the following:</p> <p>DESCENDING – Sorts identifiers in descending order of value of the variable specified. This is the default sorting order.</p> <p>ASCENDING – Sorts identifiers in ascending order of value of the variable specified.</p>
FOR <i>n</i>	Limits the report to the <i>n</i> identifiers in each time that would appear first in the report. For reports sorted in descending order, these are the <i>n</i> most active identifiers; for reports sorted in ascending order, these are the <i>n</i> least active identifiers. <i>n</i> must be an integer.
CHAR1(<i>x</i>)	Sets the plot character. Replace <i>x</i> with any valid EBCDIC character.

GRAF2 Command

Only a brief description of the GRAF2 command is given here.

Use the GRAF2 command to produce one or more sub-graphs for each individual resource included by the GRAF command. These sub-graphs can do either of the following:

Graph the performance of a second, related set of resources. For example, if your GRAF command graphs by terminal, a GRAF2 command could list each transaction used by each terminal and show the average terminal response time for each.

Graph the values of one or more variables for each resource. For example, if your GRAF command graphs by transaction, a GRAF2 command could show the usage, CPU use, and average lifetime of each transaction.

You cannot use the GRAF2 command without the GRAF command.

Command Syntax and Operands

```
GRAF2 [VARS] [idoption(identifiers)] variable [options] [HDR(text)] [CHAR1(x)]  
[SCALE({n},{mX},{COMMON})] [SORT order] [FOR n] [AND, [variable [options] [HDR(text)] [CHAR1(x)]]  
[AND, ...]
```

Operand	Meaning
VARs	Indicates that there is more than one variable in this GRAF2 report.
<i>idoption (identifiers)</i>	Any ID option that is valid for the variable, specified with or without identifiers. Specifying this operand causes the resulting sub-graphs to be organized by the specified ID option and limited to the identifiers specified by <i>identifiers</i> . If you do not specify an ID option, all identifiers of the type associated with the variable are included in the sub-graphs. You can use this operand only if you specify a single variable with the GRAF2 command.
<i>variable</i>	Any report writer variable.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.
HDR(<i>text</i>)	Overrides the default heading for the variable. Replace <i>text</i> with a heading of 15 or fewer characters.
CHAR1(<i>x</i>)	Sets the plot character for the variable. Replace <i>x</i> with any valid EBCDIC character.
SCALE(<i>n</i>) or SCALE(<i>mX</i>) or SCALE(<i>n,mX</i>) or SCALE(COMMON)	<p>By default, the highest value of a bar graph's axis is set to the highest value of the graphed variables. For variables that represent percentages, the default maximum value is 100%.</p> <p>To override this dynamic scale, specify <i>n</i> as the maximum value. If the variable that you choose produces integers, <i>n</i> must be an integer. You cannot use more than three digits to the right of the decimal point when specifying <i>n</i>.</p> <p>In some cases, bar graphs in a graph report will be too short to appear. To display these graphs, you can cause each plot character to appear as <i>m</i> characters by specifying the multiplier <i>mX</i>, where <i>m</i> is an integer. Since the resulting bar graphs will no longer correspond numerically to the scale, this is useful only for determining relative performance.</p> <p>To have the same scale used for both variables, specify SCALE(COMMON).</p> <p>You can specify the SCALE operand only with the first variable you specify with the GRAF2 command.</p>

Operand	Meaning
<code>SORT <i>order</i></code>	<p>Sorts the report. Replace <i>order</i> with one of the following:</p> <p>DESCENDING – Sorts identifiers in descending order of value of the variable specified. This is the default sorting order.</p> <p>ASCENDING – Sorts identifiers in ascending order of value of the variable specified.</p> <p>You can specify the SORT operand only with the first variable you specify with the GRAF2 command.</p>
<code>FOR <i>n</i></code>	<p>Limits the report to the <i>n</i> identifiers in each time that would appear first in the report, where <i>n</i> is an integer. For reports sorted in descending order, these are the <i>n</i> most active identifiers; for reports sorted in ascending order, these are the <i>n</i> least active identifiers. You can specify the FOR operand only with the first variable you specify with the GRAF2 command.</p>
<code>AND,</code>	<p>Signals that another variable will be specified. The comma after AND lets you continue to the next line. If you specify multiple variables, you must use the VARS operand.</p>

The following commands produce a report showing the maximum lifetime for the 10 transactions with the longest maximum lifetimes each day, and the 20 highest maximum terminal response times for each:

```
GRAF TRANSACTION MAXLIFE FOR 10 SCALE(10.0)
GRAF2 TERMINAL MAXRESP FOR 20 SCALE(10.0)
EACH DAY
END
RUN
```

GROUP Command

The GROUP command lets you combine resources into performance groups. You can use GROUP with the TAB command, the GRAF command, or any of the plot commands.

Command Syntax and Operands

```
GROUP idoption group-name(identifiers) [INCLUDE] [RANGE(rangeid)]
```

Operand	Meaning
<i>idoption</i>	An ID option identifying the type of resources to be included in the group being defined. For example, specify the ID option TRANID if the group is to consist of transactions defined by transaction IDs.
<i>group-name</i> (<i>identifiers</i>)	<p>Any 1- to 44-character name. If you use blanks in the group name, you must enclose the group name in single quotes. Replace <i>identifiers</i> with the identifiers to be included in the group (for example, transaction IDs), separated by commas. You can use generic characters in the identifiers.</p> <p>Once you define a group, you can use the group name as you would any predefined report writer ID option. That is, you can use a group name together with commands or variables to do the following:</p> <ul style="list-style-type: none">■ Organize the data in a tabular or graph report by group. For example, if you define a set of transactions with the group name MYTRANS, you can specify MYTRANS as an ID option with the TAB or GRAF commands to produce a report on just those transactions.■ Limit the value of a variable to the activity of one or more members in the group (for example, to the transactions in the group PAYROLLTRANS beginning with AC or to all transactions in the group).
INCLUDE	Includes this group in the report even if it is not included with the identifiers specified with the TAB command. The value for this group is not included in the average or total value.
RANGE (<i>rangeid</i>)	Tells the report writer to include in the group only those resources that fall within the range <i>rangeid</i> . Replace <i>rangeid</i> with the ID assigned with the RANGE command that assigns the range criteria that you want members of the defined group to meet.

The following commands define groups by terminal ID and produce a report showing the average response time of each group:

```
GROUP TERMID DIVISION.WESTERN(L1*)
GROUP TERMID DIVISION.EASTERN(L2*)
GROUP TERMID DIVISION.NORTHERN(L3*)
GROUP TERMID DIVISION.SOUTHERN(L4*)
```

```
TAB TERMID(DIVISION*) TERM RESP
END
RUN
```

The following commands define groups by ranges of transaction lifetime and produce a report showing the average use and lifetime of each group:

```
GROUP TRANID RANGE:00.000-10.000(*) RANGE(TLIFE1)
GROUP TRANID RANGE:10.001+(*) RANGE(TLIFE2)
```

```
RANGE TRAN LIFE 0-10 ID TLIFE1
RANGE TRAN LIFE 10-99999 ID TLIFE2
```

```
TAB TRANID(RANGE*) TRAN USE AND TRAN LIFE
END
RUN
```

HEADER Command

The HEADER command specifies a title to appear in a tabular report above the column headings and to the right of the PERIOD field.

The HEADER command is valid only with tabular reports.

You can specify multiple HEADER commands for one tabular report.

Command Syntax and Operands

```
HEADER(nn,mm) text
```

Operand	Meaning
<i>nn</i>	The print column where you want the first character of your heading to appear. Since headings appear on the same line as the PERIOD field, <i>nn</i> must be 20 or greater.
<i>mm</i>	The length of the heading.
<i>text</i>	The heading you want to appear in your report. Do not use string delimiters.

Related Variable Option

You can use the RIGHT variable option, described on page 10.21, to shift variable columns to the right so they do not begin before print column 20, where headings controlled by the HEADER command can begin.

The following commands create a report showing total transaction statistics, plus statistics on two particular transactions:

```

HEADER(28,13) -----EXPC-----
HEADER(46,13) -----CSSN-----

TAB TIME TRAN USE                AND,
      TRAN LIFE                AND,
      TRAN USE TRANID(EXPC) AND,
      TRAN LIFE TRANID(EXPC) AND,
      TRAN USE TRANID(CSSN) AND,
      TRAN LIFE TRANID(CSSN)
FROM 02/03/00 08:00:00
TO   02/03/00 13:00:00
EACH HOUR
END
RUN
    
```

The preceding commands create a report like the following:

SHIFT	NONE	-----EXPC-----		-----CSSN-----	
EACH	HOUR	TRAN	TRAN	TRAN	TRAN
PERIOD	NONE	USE	LIFE	USE	LIFE
TIME	TRAN	TRAN	TRAN	TRAN	TRAN
	USE	LIFE	USE	LIFE	LIFE
08.00	97	144.3	37	0.233	11 0.153
09.00	69	176.4	22	0.343	5 0.012
10.00	143	51.17	109	0.070	4 0.494
11.00	93	90.58	29	0.531	21 0.071
12.00	41	178.1	9	0.688	1 0.025
PERIOD	443	111.1	206	0.220	42 0.125

HPlot Command

Only a brief description of the HPlot command is given here. For detailed descriptions of the HPlot command, see the chapter titled “Plot Reports.”

The HPlot command plots the distribution of one variable based on the value of another variable. The variable you specify with the HPlot command is plotted along the horizontal axis. The second variable, specified with either the VPlot or the MPlot command, is plotted along the vertical axis.

You cannot use the HPlot command without either the MPlot or VPlot command. The HPlot command must follow the MPlot or VPlot command.

To set the plot character for an HPlot report, specify the CHAR1 operand with the MPlot or VPlot command.

Command Syntax and Operands

HPlot *variable* [*options*] [SCALE(*n*)] [SCATTER]

Operand	Meaning
<i>variable</i>	Any report writer variable.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.
SCALE(<i>n</i>)	<p>By default, the highest value of a plot’s horizontal axis is set to the highest value of the graphed variables. For variables that represent percentages, the default maximum value is 100%.</p> <p>To override this dynamic scale, specify <i>n</i> as the maximum value. If the variable that you choose produces integers, <i>n</i> must be an integer. You cannot use more than three digits to the right of the decimal point when specifying <i>n</i>.</p> <p>If you use the RECORD option of the EACH command, you must use SCALE(<i>n</i>), specifying a value for <i>n</i> appropriate for the activity you are plotting.</p>
SCATTER	Prints a scatter plot showing how many times each point in the plot is reached.

The following commands plot transaction use along the vertical axis against terminal response time along the horizontal axis, for each 15-minute interval:

```
VPLOT TRAN USE CHAR1(+)  
HPLOT TERM RESP SCALE(3.000)  
  EACH 15 MINUTES  
END  
RUN
```

IF Command

The IF command lets you assign new values to identifiers or create new identifiers based on existing identifiers.

Command Syntax and Operands

```
IF idoption1=(ident[,ident,ident...]) THEN idoption2=newval [ELSE  
idoption3=newval]
```

Operand	Meaning
<i>idoption1</i>	The ID option whose value is being tested.
<i>ident</i>	An ID option, or one or more identifiers to be compared to <i>idoption1</i> .
<i>idoption2</i>	The ID option whose identifiers are to be assigned a new value if the IF clause is true. This can be the same as <i>idoption1</i> .
<i>newval</i>	The new value to be assigned to the identifier. Replace <i>newval</i> with either an ID option or a character string. If you specify a character string, enclose it in single quotes.
<i>idoption3</i>	The ID option whose identifiers are to be assigned a new value if the IF clause is false. This can be the same as <i>idoption2</i> .

The following command replaces the value of TERMID with the value of TRANID whenever the value of TERMID is equal to L*:

```
IF TERMID=(L*) THEN TERMID=TRANID
```

The following command replaces the value of TRANID with the value, production, whenever the value of TRANID begins with the letter P:

```
IF TRANID=(P*) THEN TRANID='production' ELSE TRANID='test'
```

The following command creates the new ID option DEPARTMENT. Whenever the value of TRANID begins with the letter P, DEPARTMENT is assigned the value, production.

```
IF TRANID=(P*) THEN DEPARTMENT='production'
```

INCLUDE Command

The INCLUDE command defines time ranges to be included in a report.

You cannot use both INCLUDE and EXCLUDE commands in the same report. You can use multiple INCLUDE commands to include multiple time ranges.

Command Syntax and Operands

```
INCLUDE [day] [time - time] [ID name]
        [date]
        [date time - date time]
        [month]
```

Operand	Meaning
<i>day</i>	The day of the week (SUNday, MONday, TUESday, WEDnesday, THURsday, FRIday, or SATurday)
<i>time</i>	The time of day in <i>hh:mm:ss</i> format
<i>name</i>	An identifying name to be associated with a specific setting of the INCLUDE command, for use with the SELECT variable option.
<i>date</i>	The date in <i>mm/dd/yy</i> format
<i>month</i>	The month (JANuary, FEBruary, MARch, APRil, MAY, JUNe, JULy, AUGust, SEPTember, OCTober, NOVember, or DECember)

The following are examples of the INCLUDE command:

```
INCLUDE SATURDAY
```

```
INCLUDE SUNDAY
```

```
INCLUDE 07/04/00
```

```
INCLUDE 12:00:00 - 13:00:00
```

```
INCLUDE FRIDAY 16:00:00 - 17:00:00
```

```
INCLUDE 08:00:00 - 15:59:59 ID 1STSHIFT
```

```
INCLUDE 16:00:00 - 23:59:59 ID 2NDSHIFT
```

```
INCLUDE 00:00:00 - 07:59:59 ID 3RDSHIFT
```

```
INCLUDE JULY
```

INPUT Command

The INPUT command communicates three pieces of information:

- Whether the report writer is to read the performance data from the flashback file, the archive file, or the log file
- The names of the files you need to access in order to write your reports
- How many input files you want to read for one run

Command Syntax and Operands

```
INPUT(filename[(options)])
```

Operand	Meaning
<i>filename</i>	The filenames of the input files as specified in the TLBL or DLBL statements. Special input filenames are described below.
<i>options</i>	Use one or more of the following:
SYS <i>nnn</i>	SYS number assigned to this input file. The default is SYS004.
UNBLOCKED	Records are not logically blocked. Records in all Unicenter CA-Explore for CICS log files are logically blocked.
UNTIMED	Records do not contain a <i>date - time</i> field.

Default

If you use the INPUT command without a filename, or if you specify a filename using the INPUT command that does not match any name defined to the report writer, the report writer expects input from tape, and processes the file as if you had specified the filename TAPEIN.

Input Files and Descriptions

The following table describes special input filenames for Unicenter CA-Explore for CICS:

Input File	File Description
DISKIN	Unicenter CA-Explore for CICS Version 4 and higher disk log file. This filename is provided for naming a disk log file.
DISKIN x	Unicenter CA-Explore for CICS Version 4 and higher disk log files. These filenames are provided for naming additional disk log files. Replace x with any alphanumeric character.
GSIHIST	Unicenter CA-Explore for CICS Version 3 VSAM disk file. If you never had this version of Unicenter CA-Explore for CICS, you never need to specify this input file.
EXPCARC	The Unicenter CA-Explore for CICS archive file.
EXPCFBK	The Unicenter CA-Explore for CICS Version 4 and higher flashback file.
EXPCFIL	The Unicenter CA-Explore for CICS online work file.
EXPC xxx	These filenames are provided in case you have multiple masters writing to two different sets of flashback files. Replace xxx with any alphanumeric characters.
TAPC3 xx	Unicenter CA-Explore for CICS Version 3 tape log files. These filenames are provided for running reports from multiple log files of this product version. If you never had this version of Unicenter CA-Explore for CICS, you never need to specify these filenames.
TAPEIN	Unicenter CA-Explore for CICS Version 4 and higher tape log file.
TAP $xxxx$	Unicenter CA-Explore for CICS Version 4 and higher tape log files. These filenames are provided for naming additional tape log files. This lets you run reports from multiple log files in the same run. Replace $xxxx$ with any alphanumeric characters.

Input File	File Description
xxxxxxx	<p>If you specify a filename using the INPUT command that does not match any name defined to the report writer, the report writer expects input from tape and processes the file as if you had specified the filename TAPEIN.</p> <p>Do not use the following names for Unicenter CA-Explore for CICS tape log files:</p> <ul style="list-style-type: none"> ■ DISKxxx ■ EXPCxxx ■ EVSExxx ■ GSIHIST ■ TAPPWRx ■ TAPVxxx

WARNING! Since the record formats of EXPCFIL (online Unicenter CA-Explore for CICS work file) are different from the EXPCFBK and EXPCARC (Unicenter CA-Explore for CICS History Flashback and Archive files), do **not** combine EXPCFIL and EXPCFBK/EXPCARC while running batch reports.

VSE Input Files

If you run both Unicenter CA-Explore for CICS and Unicenter CA-Explore for VSE, you can produce tabular and plot reports containing data on both CICS and VSE. You can mix CICS and VSE variables **only** in reports generated by the TAB command or one of the plot commands.

To use both CICS and VSE variables, you must use the INPUT command to tell the report writer to read input data from both Unicenter CA-Explore for CICS files and Unicenter CA-Explore for VSE files. Include a DLBL or TLBL statement for each flashback or log file that you want to use as input.

See the chapter titled “Tabular Reports” for an example showing how to create a tabular report using both CICS and VSE variables.

For information about valid Unicenter CA-Explore for VSE input files, refer to the Unicenter CA-Explore for VSE *History Reporting Guide*.

INVOICE Command

The INVOICE command produces a tabular invoice report using the cost factors you specify. A separate invoice will be created for each identifier included in the report. Each invoice will begin on a new page and will include a row for each variable you specify, showing the value of the variable for the identifier and the cost associated with it.

When using the INVOICE command, you must use the COST command to specify a cost factor for each variable you want included in the invoice. Then, following the INVOICE command, specify each variable twice, once without the COST option and once with it.

If you want the report to include a summary page showing the total value and cost for each variable for all identifiers in the report, specify the variable REPORT COST TOTAL as the last variable with the INVOICE command.

Command Syntax and Operands

INVOICE [idoption[(*identifiers*)] variable [options] [HDR(*text*)] [SORT *order*]
[AND,variable [options] [HDR(*text*))] [AND,...]

Operand	Meaning
<i>idoption</i> (<i>identifiers</i>)	Any ID option that is valid for the variables you specify. You can specify one or more identifiers (separated by commas) to limit the report to a subset of the identifiers associated with the ID option. You can use generic characters when specifying <i>identifiers</i> .
<i>variable</i>	Any report writer variable. Each invoice in the report will include a row for each variable you specify.
<i>options</i>	Any valid variable options. To get complete invoices, you must specify each variable once without the COST option and once with it.
HDR(<i>text</i>)	Overrides the default description for the row for a variable. Replace <i>text</i> with the new description.
SORT <i>order</i>	Sorts the report. Replace <i>order</i> with one of the following: DESCENDING – Variables are sorted in descending order starting with the variable the SORT operand is specified with, and, when the value is the same for two entries, from left to right. This is the default sorting order. ASCENDING – Variables are sorted in ascending order starting with the variable the SORT operand is specified with, and, when the value is the same for two entries, from left to right. ALPHA – Variables are sorted in alphanumeric order with respect to the specified identifier (for example, by transaction or terminal).
AND,	Signals that another variable will be specified. The comma after AND lets you continue to the next line.

The following commands create an INVOICE report containing one page for each of the transaction identifiers. Because the variable REPORT COST TOTAL is specified, the report also includes a summary page showing the total costs for all transactions.

```

INPUT(TAPEIN)
COUNT 400
COST TRAN USE .01
COST TRAN TCPU .25
COST FILE USE .001
COST FILE ADDS .005
INVOICE TRANID TRAN USE AND,
                TRAN LIFE AND,
                TRAN TCPU AND,
                FILE USE AND,
                FILE ADDS AND,
                TRAN USE COST AND,
                TRAN TCPU COST AND,
                FILE USE COST AND,
                FILE ADDS COST AND,
                REPORT COST TOTAL

END
RUN
    
```

The following is one page of the report created by the preceding commands:

DESCRIPTION	CINQ	
	VALUE	CHARGE
TRANSACTION COUNT	105	\$1.05
TRANSACTION AVERAGE LIFETIME	1.137	
TRANSACTION TOTAL CPU TIME	3.944	\$0.99
FILE REQUEST COUNT	4107	\$4.11
FILE ADD COUNT	1	\$0.01
		\$6.16

LINECOUNT Command

The LINECOUNT command sets the number of printed lines allowed per report page.

The LINECOUNT command is a global command. Entering it once sets the line count for all reports. To avoid printing across page creases, LINECOUNT should be one of the first commands in your list of commands.

Command Syntax and Operand

LINECOUNT *n*

Operand	Meaning
<i>n</i>	The maximum number of printed lines per page. The minimum value is 20, and the maximum value depends on the size of the paper. The default is 60.

To set your line count at 72 lines per page, use the following command:

LINECOUNT 72

MPlot Command

Only a brief description of the MPlot command is given here.

The MPlot command produces a vertical plot similar to that produced by the VPlot command, but only the outline of the detail lines is printed. You can specify as many variables per MPlot report as you want.

Command Syntax and Operands

M PLOT *variable* [*options*] [SCALE(*n*),*mX*)] [CHAR1(*x*)] [HDR1(*text*)] [HDR2(*text*)] [HDR(*text*)] [AND,
variable [*options*] [SCALE(*n*),*mX*)] [CHAR1(*x*)] [HDR1(*text*)] [HDR2(*text*)] [HDR(*text*)] [AND,...]

Operand	Meaning
<i>variable</i>	Any report writer variable.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.
SCALE(<i>n</i>), or SCALE(<i>mX</i>) or SCALE(<i>n,mX</i>)	By default, the highest value of a plot's horizontal axis is set to the highest value of the graphed variables. For variables that represent percentages, the default maximum value is 100%. To override this dynamic scale, specify <i>n</i> as the maximum value. If the variable that you choose produces integers, <i>n</i> must be an integer. You cannot use more than three digits to the right of the decimal point when specifying <i>n</i> . You can also widen the outline of the plot by using the multiplier <i>mX</i> , where <i>m</i> is an integer.
CHAR1(<i>x</i>)	Sets the plot character. Replace <i>x</i> with any valid EBCDIC character.
HDR1(<i>text</i>)	Overrides the default top heading for the scale of the plot. Replace <i>text</i> with the new heading. To remove the header, replace <i>text</i> with a null string, as in HDR1().
HDR2(<i>text</i>)	Overrides the default bottom heading for the scale of the plot. Replace <i>text</i> with the new column heading. To remove the header, replace <i>text</i> with a null string, as in HDR2().
HDR(<i>text</i>)	Overrides the default description beside the SYMBOLS: heading that describes the plotted variables. Replace <i>text</i> with the new description, not to exceed 30 characters.
AND,	Signals that another variable will be specified. The variable will be plotted on top of the preceding variables. The comma after AND lets you continue to the next line.

OPTION Command

The OPTION command defines cosmetic options for all reports.

The OPTION command is a global command. Entering it once sets options for all reports.

Command Syntax and Operands

```
OPTION ([AVERAGE=[NO | YES | n ]
        [DEFERRED | DNW | IMMEDIATE]],
        [CASE=UPPER],
        [DATAONLY=[NO | YES]],
        [DATEFRMT=[MM/DD/YY | DD/MM/YY]],
        [DEFEREXT=[NO | YES]],
        [DELAY=nnr%, DINT=mm],
        [DUMP=[NO | YES]] [CORRUPT],
        [ECHO=[YES | NO]],
        [HEXID=[NO | YES]],
        [IOERR=[NO | YES | BYPASS]],
        [JOBZONE=[nWEST | nEAST]],
        [MASTER=jobname],
        [MAXTO=[YES | NO]],
        [MONETARY=n],
        [NULLID=[BLANK | 'xxxx' | n/a]],
        [NULLLINE=[YES | NO]],
        [PERIOD=WEIGHTED]],
        [PGMSTOR=[NO | YES]].
        [PRODUCT=EVSE|EXPC]
        [RANGE=WEIGHTED],
        [RECAP=[NO | YES],
        [RECORD=n]
        [RECSTAT=[NO | YES]],
        [REWIND=UNLOAD],
        [RPTSTAT=[NO | YES | xxxxx]],
        [STORAGE=[BELOW | ABOVE]],
        [STORSTAT=[NO | YES],
        [SUMMARY=YES | NO | n],
        [TAB2SKIP=[n | PAGE]
        [TRACE=[NO | STORAGE | LOADS],
        [WEEK=weekday]
        [XMSG=[nnn | ALL],
        [ZEROFD=[BLANK | ZERO | 'xxxx']]])
```

Operand	Meaning	Default
AVERAGE	<p>Specifies whether a line giving averages of the items listed in the report is to be printed. If you set AVERAGE=<i>n</i>, the averages are printed only if the report contains <i>n</i> or more detail lines.</p> <p>For graph reports, you can specify when the average is calculated and if it is weighted. AVERAGE=IMMEDIATE calculates the average line during data accumulation. The average is weighted. AVERAGE=DEFERRED calculates the average line just prior to printing the report. The average is weighted. This setting is useful when you use the FOR operand or RANGE DEFERRED command and want the average line to reflect only the lines being printed. AVERAGE=DNW (Deferred Not Weighted) calculates the average line just before printing the report, but the average is not weighted.</p>	<p>NO (no average line)</p> <p>For graph reports, IMMEDIATE</p>
CASE	<p>Specify CASE=UPPER to force all output to upper case. You can also use the PARM= <i>parameter</i> of your // EXEC statement to perform this function.</p>	<p>Output is in mixed case.</p>
DATAONLY	<p>Specify YES to print only lines containing report data. Otherwise, all lines are printed. You can also use the PARM= <i>parameter</i> of your // EXEC statement to perform this function.</p>	<p>NO</p>
DATEFRMT	<p>Sets the format of dates in the report output. You can combine the following elements in any order:</p> <ul style="list-style-type: none"> ■ YY or YYYY for year ■ MM for month ■ DD for day <p>Characters other than those listed above are assumed to be separators and appear as entered. For example, YYYY.MM.DD displays a four-digit year, a two-digit month, and a two-digit day separated by periods.</p> <p>You can also use the PARM= <i>parameter</i> of your // EXEC statement to perform this function.</p>	<p>MM/DD/YY</p>

Operand	Meaning	Default
DEFEREXT	Specifies when a user exit is to be called. DEFEREXT=NO passes control to the exit before any processing is done with the current record. DEFEREXT=YES defers the user exit call until after all selection criteria have been applied and passes to the user exit only those records passing all selection criteria (such as ID options and RANGE commands).	NO
DELAY	Delays report processing to lessen the impact of the CPU-intensive report writer. Replace <i>mm</i> with the percentage of time you want to delay processing. Replace <i>mm</i> with the interval, in seconds, for which the percentage is calculated. For example, if you specify the following, OPTION(DELAY=50%,DINT=30) report writer processing will be delayed 15 seconds every 30 seconds.	DELAY=0% and DINT=30
DUMP	Specifies whether to print a diagnostic dump of input records. DUMP=YES dumps all used records. DUMP=CORRUPT dumps all records with detected corruption.	NO
ECHO	Specifies whether to list your report commands before printing the report. Set the value to NO only after you verify the report commands are correct. Any report commands preceding the ECHO=NO command in the job stream will appear with the report. You can also use the PARM= <i>parameter</i> of your // EXEC statement to perform this function.	YES
HEXID	Specifies whether to allow nonprintable characters in identification fields. HEXID=NO treats any identification field that contains a nonprintable character as invalid. All information regarding the resource is saved and printed under the identifier INVALID. HEXID=YES deactivates the test for nonprintable characters. When the report is printed, however, all nonprintable characters are printed as periods.	YES

Operand	Meaning	Default
IOERR	Determines what happens if the report writer encounters an I/O error in input data. If IOERR=NO, the report writer stops and no reports are printed. IOERR=YES causes the report writer to treat the error as an end-of-file and produces reports defined up to that point. If IOERR=BYPASS, the report writer will attempt to bypass the error condition and continue processing input.	NO
JOBZONE	Sets the job zone (the number of hours east or west of Greenwich Mean Time). Specify <i>n</i> EAST or <i>n</i> WEST, where <i>n</i> is an integer.	Value set by the ZONE parameter of the IPL SET command
MASTER	Include this operand when you use archive file input to insure that the report writer reads the most recent data logged. Replace <i>jobname</i> with the jobname of the master region.	No default
MAXTO	Applies only if every report in the run includes a TO command. By default, an end-of-file condition is forced when the maximum TO time specified with any report is exceeded. If the input file is not in chronological order, specify MAXTO=NO to force the entire file to be read.	YES
MONETARY	Specifies the character to precede currency numbers. You can specify any single character.	\$ (dollar sign)
NULLID	Defines a four-character string to be printed in any identifier field for which no identifier exists. Specify NULLID=BLANK to print blanks in fields for which no identifier exists.	The characters n/a
NULLINE	Specifies whether lines with all zero values should print.	YES for plot reports NO for tabular reports
PERIOD	For a plot report, controls whether the period line is a weighted value; that is, the average of all the detail lines. PERIOD=WEIGHTED converts the period line into a weighted value that represents a true average over the period. For a tabular report, controls whether period lines are printed.	Not weighted for plot reports. YES for tabular reports.
PGMSTOR	Controls whether the diagnostic storage analysis facility displays the storage acquired by each report writer internal program.	NO

Operand	Meaning	Default
PRINT	<p>For tabular reports only, specify OTHERS to insert a line above the summary line that represents all the data excluded by the FOR operand.</p> <p>For tabular reports only, specify GROUP to restrict the report output to the groups specified by the GROUP command</p>	No default
PRINTLEN	Specifies the printed line length.	133
PRODUCT	Specifies the Unicenter CA-Explore for CICS product for DEFINE XID or DEFINE XVAR commands. Specify EXPC.	No default
RANGE	Controls whether the range line of a plot report is a weighted value; that is, the average of all the detail lines. RANGE=WEIGHTED converts the range line into a weighted value.	Not weighted
RECAP	Specifies whether to produce a summary value for each identifier in the report. Depending on the variable, the summary line includes the total, average, maximum, or minimum values of the variable. This option is valid only for tabular reports organized by identifier and time.	<p>NO</p> <p>If OUTPUT (RVIEW) is specified, then the default is YES.</p>
RECORD	Specifies the decimal value of the record type for DEFINE XID and DEFINE XVAR commands.	No default
RECSTAT	<p>Specifies whether to print statistics about Unicenter CA-Explore for CICS performance, accounting, and threshold records following your report.</p> <p>RECSTAT=YES prints information including the length of each record and each of its components (in the case of performance records), the percentage of the total disk space that each record or component comprises, and the time period covered by each. This information can help you determine which CICS resources Unicenter CA-Explore for CICS should collect data about. In particular, if a given component contains a great deal of data, but you seldom run reports on this data, you may choose to stop collecting that data in order to lessen the storage required for the flashback and archive files.</p> <p>For similar tuning information organized by transaction rather than by record type, see the description of the CICS.PERFORMANCE.RECORD.ANALYSIS.STAT canned report.</p>	NO

Operand	Meaning	Default
REWIND	Applies only when a single tape input file has been specified with the INPUT(<i>xxxxxxx</i>) command. REWIND=UNLOAD specifies that the input tape is to rewind and unload when it reaches an EOF (end-of-file) or EOY (end-of-volume) condition. If you have specified multiple tape input files, all tapes automatically rewind and unload.	Rewind only
RPTSTAT	Controls how often the following message is issued to the console: ERPT274I STATUS: CNT - <i>nnnnn</i> . TIME - <i>mm/dd/yy hh:mm:ss</i> <i>nnnnn</i> is the current number of input records processed. <i>mm/dd/yy hh:mm:ss</i> is the timestamp of the current record. RPTSTAT=YES produces a status message every 10,000 records. RPTSTAT= <i>nnnnn</i> produces a status message every <i>nnnnn</i> records. You can also specify HOUR, DAY, or WEEK to generate the status message on an hourly, daily, or weekly interval.	NO (no such messages are issued)
STORAGE	If 31-bit addressing is available, specifies whether storage is acquired above or below the 16M line.	ABOVE
STORSTAT	Specifies whether to print storage statistics following the last report. STORSTAT=YES prints the storage amounts allocated to each report and the total amount of unused virtual storage.	NO
SUMMARY	Pertains only to tabular reports using an ID option. SUMMARY=YES prints a summary line. If you specify SUMMARY= <i>n</i> , summary lines are printed only if the report contains at least <i>n</i> lines. Depending on the variable, the summary line includes the total, average, maximum, or minimum values of the variable.	YES
TAB2SKIP	For TAB2 reports only, controls how much space is left between identifiers listed by the TAB command. Either specify TAB2SKIP= <i>n</i> , where <i>n</i> is the number of lines to be skipped, or specify TAB2SKIP=PAGE to generate a page break after each identifier.	0 (no space)

Operand	Meaning	Default
TRACE	Controls the diagnostic tracing facility. TRACE=STORAGE causes message ERPT267I to print every time an internal storage request is processed. This message shows the name of the internal program requesting storage and shows how much storage the program is requesting. TRACE=LOADS causes message ERPT283I to be printed whenever a report writer module is loaded or freed. This message shows the name of the module.	NO
WEEK	Specifies the first day of the week (SUNday, MONday, TUESday, WEDnesday, THURsday, FRIday, or SATurday). If you specify the EACH <i>nnn</i> WEEKs command, you can use this operand to specify whether a week is Monday through Sunday, Tuesday through Monday, and so on.	MONday
XMSG	Excludes messages from printing and from displaying on the console. XMSG= <i>nnn</i> excludes message ERPT <i>nnn</i> from output. XMSG=ALL excludes all messages.	No messages are excluded
ZEROFLD	Defines a value to replace any zero values. ZEROFLD=BLANK replaces any zero value with a blank. ZEROFLD=ZERO prints a zero. ZEROFLD= <i>xxx</i> replaces each zero value with <i>xxx</i> .	BLANK for tabular reports ZERO for graph and plot reports

This is a sample OPTION command:

```
OPTION(ZEROFLD=ZERO, NULLLINE=YES, AVERAGE=5)
```

This command performs the following actions:

- Prints a zero for each zero value in the report
- Prints lines with all zero values
- Prints a line showing the average values of the items in the report if the report contains five or more detail lines

OUTPUT Command

Use the OUTPUT command to send report output to a disk or tape file that you specify. When you use the OUTPUT command, you must include additional statements in your JCL.

Output to a Disk File

If your report output is going to a disk file, include the following statements in your JCL:

```
//ASSGN SYS005,DISK,VOL=volser,SHR  
//DLBL filename
```

Replace *volser* with the volume ID of your disk device, and replace *filename* with the output file name.

Output to a Tape File

If your report output is going to a tape file, include the following statements in your JCL:

```
//ASSGN SYS005,cuu  
//TLBL filename
```

Replace *cuu* with the CUU of your tape device, and replace *filename* with the output file name.

Command Syntax and Operands

OUTPUT{(*filename* , [SYS*nnn*] , [DATA] , [DISK|TAPE] , [NOHDR] , [NODICT])}

Operand	Meaning
<i>filename</i>	An output TLBL or DLBL name. Specify <i>filename</i> without any other operands to send your report output to a sequential output disk file.
SYS <i>nnn</i>	The SYS number corresponding to the ASSGN and EXTENT cards. The default is SYS005.
DATA	Converts the report output into formatted output records, which can be processed by a non-CA product. This operand is valid only for tabular reports.
DISK	Specifies that the output file is to be on disk. This is the default. This operand is valid only for tabular reports.
TAPE	Specifies that the output file is to be on tape. This is the default. This operand is valid only for tabular reports.
NOHDR	Indicates to exclude the Unicenter CA-Explore for CICS standard SMF record header. This operand is valid only for tabular reports.
NODICT	Indicates to exclude the dictionary record. This operand is valid only for tabular reports.

This is an example of the OUTPUT command in which the report output is written to the disk file TRANUSE:

```
//ASSGN SYS000,DISK,VOL=volser,SHR
//DLBL TRANUSE
//EXTENT SYS000,volser,1,0,1008,27
//LIBDEF*,SEARCH=(EXPLORE.SUBLIB)
//EXEC EXPRPT,SIZE=EXPRPT
INPUT(EXPCFBK)
OUTPUT (TRANUSE,SYS000,DATA,NOHDR,NODICT)
TAB USERID TRAN USE
END
RUN
```

PERIOD Command

The PERIOD command produces one report per period, including a summary of the **average** activity of variables during a specified time. It can be used with the PLOT and TAB commands.

You must have the appropriate amount of data in your input file for the time you specify. For example, you cannot specify PERIOD YEAR if your input file has less than one year's data.

Command Syntax and Operands

```
PERIOD time-period [SKIP=n]
```

Operand	Meaning
<i>time-period</i>	<p>One of the following base time units. <i>nnn</i> shown below is the number of time intervals (up to 999) in the time unit you select. The default is 1.</p> <ul style="list-style-type: none"> ■ MINUTE ■ <i>nnn</i> MINUTES ■ HOUR ■ <i>nnn</i> HOURS ■ DAY ■ <i>nnn</i> DAYS ■ WEEK ■ <i>nnn</i> WEEKS ■ MONTH ■ <i>nnn</i> MONTHS ■ YEAR ■ <i>nnn</i> YEARS
SKIP= <i>n</i>	<p>Defines the number of lines to skip between reports for each period. <i>n</i> can range from 1 to 9. If you do not specify SKIP= <i>n</i>, a page break occurs between periods.</p>

To request a summary of average daily activity for a resource, use the following commands:

```
PLOT variable
  PERIOD DAY
END
RUN
```

For a summary of average activity for a resource every three months, use the following commands:

```
TAB variable
  PERIOD 3 MONTHS
END
RUN
```

PLOT Command

Only a brief description of the PLOT command is given here. For a detailed description of the PLOT command, see the chapter titled “Plot Reports.”

The PLOT command produces reports that plot resource usage horizontally across time.

Command Syntax and Operands

```
PLOT variable [options] [ALONE] [SCALE({n}, {mX})] [CHAR1(x)]
                                         [CHAR2(x)]
```

Operand	Meaning
<i>variable</i>	Any report writer variable.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.
ALONE	Prevents the display of the global (system-wide) value of the variable. To use this operand, you must restrict the variable with an ID option.
SCALE(<i>n</i>) or SCALE(<i>mX</i>) or SCALE(<i>n,mX</i>)	By default, the highest value of a plot’s horizontal axis equals the highest value reached by a variable in the plot or, for variables representing percentages, 100%. To override this dynamic scale, specify <i>n</i> as the maximum value. If the variable that you choose produces integers, <i>n</i> must be an integer. In some cases, bar graphs in a plot report will be too short to appear. To display these graphs, you can cause each plot character to appear as <i>m</i> characters by specifying the multiplier <i>mX</i> , where <i>m</i> is an integer. Since the resulting bar graphs will no longer correspond numerically to the scale, this is useful only for determining relative performance.
CHAR1(<i>x</i>)	Sets the plot character that represents the restricted variable. Replace <i>x</i> with any valid EBCDIC character.
CHAR2(<i>x</i>)	Sets the plot character that represents the global variable. Replace <i>x</i> with any valid EBCDIC character.

The following commands plot the value of the variable TRAN LIFE for transaction Cinq and the value for all transactions, using a scale of 0 to 500. The character X will be used to plot the activity of transaction Cinq and a plus sign (+) will be used to plot the activity of all transactions.

```
PLOT TRAN LIFE TRANID(CINQ) SCALE(500) CHAR1(X) CHAR2(+)  
END  
RUN
```

PLOT2 Command

Only a brief description of the PLOT2 command is given here.

The PLOT2 command is used with the PLOT command to plot two variables across time. This lets you compare the performance of two variables side-by-side in a single report to see if a cause-and-effect relationship exists between the two.

You cannot use the PLOT2 command without the PLOT command.

Command Syntax and Operands

```
PLOT2 variable [options] [ALONE] [SCALE({n},{mX},{COMMON})] [CHAR1(x)]  
CHAR2(x)
```

The operands for the PLOT2 command are the same as for the PLOT command except for the COMMON option of the SCALE operand. Specify SCALE(COMMON) if you want the same scale to be used to plot both variables.

For an explanation of the other PLOT2 operands, refer to the descriptions of the PLOT operands earlier in this chapter.

In the following example, the PLOT and PLOT2 commands are used to compare the transaction use of transaction C1NQ against its lifetime, using the same scale for both variables. Because the ALONE operand is specified, global activity (the values of the variables for all transactions) will not be plotted.

```
PLOT  TRAN USE          TRANID(C1NQ) ALONE  
PLOT2 TRAN LIFETIME    TRANID(C1NQ) ALONE SCALE(COMMON)  
END  
RUN
```

PRTEXT Command

PRTEXT command defines a user exit to be called before each line is printed.

Supplied User Exit

The user exit PRTGUX is predefined for your use. See the appendix “User Exits” for more information.

Command Syntax and Operand

PRTEXT(*userexit*)

Operand	Meaning
<i>userexit</i>	The name of the user exit

RANGE Command

The RANGE command sets criteria that must be met before data is included in a report.

You can use the RANGE command only with report writer variables that have numeric values.

Command Syntax and Operands

```
RANGE variable [RATE] min1-max1,min2-max2,... [ALONE] [DEFERRED] [ID rangeid]  
[UNDERLINE]  
[HIGHLIGHT]
```

Operand	Meaning
variable	Any numeric report writer variable, to which the range criteria are to be applied.
RATE	Allows you to specify range values as rates (number per second). RATE is valid only when DEFERRED is specified.
<i>min1-max1, min2-max2, ...</i>	Up to 10 minimum and maximum values for the variable chosen. You can specify a single value, or a single value followed by a plus sign (+) to signify no upper boundary. When you specify a pair, records are included if the variable value you specify is greater than or equal to the minimum value and less than the maximum value.
ALONE	Applies only if multiple RANGE commands are used in a single report. If more than one RANGE command is used, all range criteria must be satisfied before a record is accepted. If one range value fails, the record is discarded. If the ALONE option is specified on one RANGE command and the record passes its range criteria, the record is accepted and all other RANGE commands are ignored.
DEFERRED	Waits to apply the range criteria until after all the input data is read, and then discards lines having values that do not meet those range criteria. For example, you would need to use the DEFERRED operand to report on only those jobs that were executed 10 or more times, since the report writer is able to determine this only after it reads all the input data. The RATE option is valid for the RANGE command when you use the DEFERRED operand.

Operand	Meaning
UNDERLINE	Like the DEFERRED operand, waits to apply range criteria until after all the input data is read. However, instead of discarding lines having values that do not meet the range criteria, the UNDERLINE option prints all lines and underlines those values that meet a range criterion.
HIGHLIGHT	Similar to UNDERLINE but prints values meeting the range criteria in boldface rather than underlining them. If the report is printed on a laser printer, these values may not appear to have been bolded; in that case, use the UNDERLINE operand instead of the HIGHLIGHT operand.
ID <i>rangeid</i>	In a tabular report, assigns an identifier to the specified range of values for the variable. <i>rangeid</i> must be 1-8 characters. You can use the GROUP command specify a group that includes only those resources that fall within the range named <i>rangeid</i> .

To plot the activity of transactions with response times under one second, use the following commands:

```
PLOT TRAN USE
  RANGE TERM RESP 0-1 DEFERRED
END
RUN
```

The following commands define groups by ranges of transaction lifetime and produce a report showing the average use and lifetime of each group:

```
GROUP TRANID RANGE:00.000-10.000(*) RANGE(TLIFE1)
GROUP TRANID RANGE:10.001+(*) RANGE(TLIFE2)

RANGE TRAN LIFE 0-10 ID TLIFE1
RANGE TRAN LIFE 10-99999 ID TLIFE2

TAB TRANID(RANGE*) TRAN USE AND TRAN LIFE
END
RUN
```

RUN Command

The RUN command ends the interpretation of commands and begins the processing and printing of reports.

The RUN command should be the last command in your job stream. Any command after the RUN command is ignored.

Command Syntax

```
RUN
```

Example

The following is an example of the RUN command in a job stream:

```
TITLE1 REPORT1
PLOT variable
  FROM 02/01/03 00.00.01
  TO   02/15/03 23.59.59
END
*
TITLE1 REPORT2
PLOT variable
  SHIFT 17:00:00 01:00:00
END
.
.
.
RUN
```

SELECT Command

Use the SELECT command to specify the identifiers to be included in or excluded from a report. This command is particularly useful in restricting the following types of reports:

- Canned reports, which cannot otherwise be restricted to specific resources; the only alternative to using the SELECT command is to use the commands in the expansion rather than the CANRPT command, tailoring the report commands as required.
- Tabular reports not organized by resource, which can otherwise be restricted only by specifying the ID option and identifiers with every variable included in the report.

Command Syntax and Operands

```
SELECT idoption(identifiers)
```

Operand	Meaning
<i>idoption(identifiers)</i>	Any ID option followed by one or more identifiers of the appropriate type. You can use generic characters when specifying identifiers. Separate multiple identifiers by commas.

The following commands generate a CICS.TERM.STAT canned report. The SELECT command limits the report to terminals having IDs that do **not** begin with the letter T.

```
SELECT TERMD(-T*)
CANRPT CICS.TERM.STAT
END
RUN
```

The following commands generate a tabular report organized by time but not by resource. The SELECT command restricts the report to three transactions, an alternative to specifying the ID option and identifiers `TRANID(PAY1,PAY2,PAY3)` with each variable.

```
SELECT TRANID(PAY1,PAY2,PAY3)
TAB TIME TRAN USE AND,
          TRAN USE RATE HDR2(RATE) AND,
          TRAN USE DIFF HDR2(%CHANGE) AND,
          TRAN I/OTIME AND,
          TRAN MAXI/OTIME AND,
          TRAN CPUTIME AND,
          TRAN ABENDS AND,
          TRAN GETMSIZE
EACH HOUR
END
RUN
```

SHIFT Command

The SHIFT command restricts the hours of the day covered by a report.

Like most secondary commands, the SHIFT command cannot be used twice for the same report, even if two different times are specified.

The following figure shows an incorrect use of SHIFT commands in a command stream and the corrected command stream:

Incorrect Command Stream	Corrected Command Stream
<pre>GRAF <i>variable</i> SHIFT 08:00:00 12:00:00 SHIFT 13:00:00 17:00:00 END RUN</pre>	<pre>GRAF <i>variable</i> SHIFT 08:00:00 17:00:00 END GRAF <i>variable</i> SHIFT 13:00:00 17:00:00 END RUN</pre>

Command Syntax and Operands

SHIFT *start-time end-time*

Operand	Meaning
<i>start-time</i>	The earliest time to include in the report, in <i>hh:mm:ss</i> format, where <i>hh</i> is hours, <i>mm</i> is minutes, and <i>ss</i> is seconds. You can use periods instead of colons to separate the hours, minutes, and seconds.
<i>end-time</i>	The latest time to include in the report, in the same format as <i>start-time</i> .

To graph only the activity that occurs between 8:00 a.m. and 5:00 p.m., enter the following:

```
GRAF variable
  SHIFT 08:00:00 17:00:00
END
RUN
```

Shifts can cross midnight. For example, to write a tabular report that covers from 11:00 p.m. to 7:00 a.m., use the following commands:

```
TAB variable
  SHIFT 23:00:00 07:00:00
END
RUN
```

TABulate Command

Only a brief description of the TAB command is given here.

The TAB command produces tabular reports. The variables you specify with the TAB command determine the columns in your report and the order in which they appear. The operands you specify determine:

- How the report is organized: by time, by classes of or individual identifiers (such as transactions, terminals, or users), or by the relationship between identifiers
- How the report is sorted
- The column headings that appear in the report

Command Syntax and Operands

```
TAB [period] [idoption[(identifiers)]] variable [options]
[HDR1(text)] [HDR2(text)] [SORT order] [FOR n] [AND,
[TOP n]
[BOTTOM n]
variable [options] [HDR1(text)] [HDR2(text)] [AND,... ]
```

Operand	Meaning
<i>period</i>	Sets the first column in the report to one of the following: DATETIME – The date and time DATE – The date TIME – The time DAY – The day (MONDAY, TUESDAY, and so on)
<i>idoption</i> (<i>identifiers</i>)	Any ID option that is valid for the variable, specified with or without identifiers. Specifying this operand causes the resulting report to be organized by the specified ID option and limited to the identifiers specified by <i>identifiers</i> . You can specify multiple ID options. Identifiers are listed either beginning in the first column (if you did not specify a time period as the first column) or beginning in the second column.
<i>variable</i>	Any report writer variable.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.

Operand	Meaning
HDR1(<i>text</i>)	Overrides the default top heading for a column. (Each column has two default eight-character lines.) Replace <i>text</i> with the new column heading. To remove the header, replace <i>text</i> with a null string, as in HDR1().
HDR2(<i>text</i>)	Overrides the default bottom heading for a column. Replace <i>text</i> with the new column heading. To remove the header, replace <i>text</i> with a null string, as in HDR2().
<i>order</i>	<p>Sorts the report. Replace <i>order</i> with one of the following:</p> <p>DESCENDING – Variables are sorted in descending order starting with the variable the SORT operand is specified with, and, when the value is the same for two entries, from left to right. This is the default sorting order.</p> <p>ASCENDING – Variables are sorted in ascending order starting with the variable the SORT operand is specified with, and, when the value is the same for two entries, from left to right.</p> <p>ALPHA – Variables are sorted in alphanumeric order with respect to the specified identifier (for example, by transaction or terminal).</p> <p>Use SORT only if the report is organized by identifier (for example, by transaction). If the report is organized by time, then it is already sorted.</p>

Operand	Meaning
FOR <i>n</i>	<p>For Reports Organized by:</p> <ul style="list-style-type: none"> ■ Identifier ■ Identifier, and time or date <p>Limits reports to the <i>n</i> identifiers in each time that would appear first in the report, where <i>n</i> is an integer. For reports sorted in descending order, these are the <i>n</i> most active identifiers. For reports sorted in ascending order, these are the <i>n</i> least active identifiers. For reports sorted in alphanumeric order, these are the <i>n</i> identifiers coming first alphanumerically.</p> <p>For Reports Organized by Time or Date:</p> <p>Limits reports to the <i>n</i> times or dates with the highest values for the first variable specified. For reports sorted by a variable in descending order, these are the <i>n</i> times or dates with the highest value for that variable. For reports sorted by a variable in ascending order, these are the <i>n</i> times or dates with the lowest value for that variable.</p>
TOP <i>n</i>	<p>Limits reports organized by identifier to the <i>n</i> identifiers with the highest values for the first variable specified, where <i>n</i> is an integer. Unlike the FOR operand, which determines which identifiers to include in the report only after all the data has been read and processed, the TOP operand tests each record as it is read and discards records that do not qualify, saving storage. The TOP operand is appropriate for ID options with many unique identifiers, such as TRANNUM.</p>
BOTTOM <i>n</i>	<p>Identical to the TOP operand except it prints the <i>n</i> identifiers with the lowest values for the first variable specified.</p>
AND,	<p>Signals that another variable will be specified. The comma after AND lets you continue to the next line.</p>

The following commands create a tabular report organized by date and time that shows the transaction use and average transaction lifetime for all transactions each hour:

```
TAB DATETIME TRAN USE    AND,
      TRAN LIFE
      EACH HOUR
END
RUN
```

The following commands create a tabular report organized by date and time that shows the five hours with the highest average transaction lifetime:

```
TAB DATETIME TRAN LIFE FOR 5
      EACH HOUR
END
RUN
```

TAB2 Command

Only a brief description of the TAB2 command is given here.

The TAB2 command builds a secondary table for each row of the table created by the TAB command.

Command Syntax and Operands

```
TAB2 idoption[(identifiers)] variable [options] [HDR1(text)] [HDR2(text)] [SORT
order]
[FOR n] [AND,
[TOP n] [BOTTOM n]
variable [options] [HDR1(text)] [HDR2(text))] [AND,...]
```

The TAB2 command uses the same operands as the TAB command, except that you cannot specify both a period operand (DATETIME, DATE, TIME, or DAY) and an ID option.

You cannot use the TAB2 command without the TAB command.

You can use the TAB2 command only if you also specify the TAB command with an ID option. You can specify a period option (DATETIME, DATE, TIME, and DAY) with either the TAB command or the TAB2 command, but *not* with both.

The following commands create a tabular report organized first by transaction ID and then by the date:

```
TAB  TRANID TRAN USE
TAB2 DATE   TRAN USE
      EACH HOUR
END
RUN
```

TITLE1, TITLE2, and TITLE3 Commands

Three 60-character lines at the top of each page are reserved for titles. The TITLE1, TITLE2, and TITLE3 commands write titles on the first, second, and third of these lines, respectively. If you do not use the TITLE commands, the report writer assigns title lines to the report.

Titles are printed only for the report in which the TITLE commands occur. Use the GLOBAL TITLE n command to print the same title for all reports in one run.

The TITLE1 title is printed first, the TITLE2 title is printed below it, and the TITLE3 title appears last. You must specify the TITLE1, TITLE2, and TITLE3 commands in that order. You can use none, one, or all of these commands in any combination.

Command Syntax and Operand

TITLE1 *titletext*

TITLE2 *titletext*

TITLE3 *titletext*

Operand	Meaning
<i>titletext</i>	<p>The title that you assign to the report; use up to 60 characters. Every character in <i>titletext</i> must be on the same line as the TITLE1, TITLE2, or TITLE3 command; you cannot continue titles with a continuation character.</p> <p>To dynamically insert dates and times into <i>titletext</i>, use the keywords described in the next section.</p>

Keywords

You can use keywords to dynamically place dates and times in your report titles. Keywords have the following format:

&SourceRangeFormat

Replace the variables as shown in the following table:

Replace	With
<i>&Source</i>	Specify one of the following to determine the origin of the dates and times: C Uses report command specifications. R Uses the starting and ending times for the data in the report. E Uses the EACH command specifications. P Uses the PERIOD command specifications.
<i>Range</i>	Specify one of the following to determine whether to take data from the FROM command or the TO command: F Uses the FROM command specification. T Uses the TO command specification.

Replace	With
<i>Format</i>	Specify one of the following to determine the format of the date and time: WEEKDAY – Monday, Tuesday, and so on DATE – <i>mm/dd/yy</i> or <i>dd/mm/yy</i> (European format) DAY – <i>dd</i> EDAY – <i>dd</i> , if your dates are in European format MONTH – <i>mm</i> EMONTH – <i>mm</i> , if your dates are in European format YEAR – <i>yy</i> TIME – <i>hh:mm:ss</i> HOUR – <i>hh</i> MINUTE – <i>mm</i> SECOND – <i>ss</i> CALMONTH – January, February, and so on JDAY – <i>ddd</i> (Julian date)

To have the title Transaction Use by Hour appear on title line 2, enter the following command:

```
TITLE2 Transaction Use by Hour
```

The following commands dynamically create a date and time on title lines 2 and 3, determined by the data in the report and the FROM and TO commands:

```
TITLE2 FROM: &RFWEEKDAY, &RFCALMONTH &RFDAY, 19&RFYEAR at &RFTIME
TITLE3 TO: &RTWEEKDAY, &RTCALMONTH &RTDAY, 19&RFYEAR at &RTTIME
```

The following is sample output for these commands:

```
FROM: Wednesday, January 27, 1999 at 02.58.00
TO: Monday, February 01, 1999 at 11.23.59
```

VPLOT Command

Only a brief description of the VPLOT command is given here.

The VPLOT command plots resource usage in vertical columns. The horizontal axis of the plot represents time.

Command Syntax and Operands

```
VPLOT variable [options] [SCALE({n},{mX})] [STACK=id] [CHAR1(x)] [HDR1(text)] [HDR2(text)] [HDR(text)]
```

Operand	Meaning
<i>variable</i>	Any report writer variable.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.
SCALE(<i>n</i>) or SCALE(<i>mX</i>) or SCALE(<i>n,mX</i>)	By default, the highest value of a plot's vertical axis equals the highest value reached by a variable in the plot or, for variables representing percentages, 100%. To override this dynamic scale, specify <i>n</i> as the maximum value. If the variable that you choose produces integers, <i>n</i> must be an integer. You can also widen each plot column by using the multiplier <i>mX</i> , where <i>m</i> is an integer.
STACK= <i>id</i>	Replaces the plot character with the identifiers being plotted, from most active to least active starting at the bottom of the plot. You can specify the <i>mX</i> option of the SCALE operand to print more characters of each identifier.
CHAR1(<i>x</i>)	Sets the plot character. Replace <i>x</i> with any valid EBCDIC character.
HDR1(<i>text</i>)	Overrides the default top heading for the scale to the left of the plot. Replace <i>text</i> with the new heading. To remove the header, replace <i>text</i> with a null string, as in HDR1().
HDR2(<i>text</i>)	Overrides the default bottom heading for the scale to the left of the plot. Replace <i>text</i> with the new heading. To remove the header, replace <i>text</i> with a null string, as in HDR2().
HDR(<i>text</i>)	Overrides the default description beside the SYMBOLS: heading that describes the plotted variable. Replace <i>text</i> with the new description, not to exceed 30 characters.

The following commands plot transaction use against time using the letter T as the plot character:

```
VPLOT TRANSACTION USE CHAR1(T)
END
RUN
```

VPLOT2 Command

Only a brief description of the VPLOT2 command is given here.

The VPLOT2 command is used with the VPLOT command to plot two variables across time. This lets you compare the performance of two variables side-by-side in a single report to see if a cause-and-effect relationship exists between them. The VPLOT2 variable is printed directly over the VPLOT variable.

You cannot use the VPLOT2 command without the VPLOT command.

Command Syntax and Operands

```
VPLOT2 variable [options] [SCALE({n},{mk},{COMMON})] [CHAR1(x)] [HDR1(text)] [HDR2(text)] [HDR(text)]
```

The operands for the VPLOT2 command are the same as for the VPLOT command except that the VPLOT2 command does not support the STACK operand and it does support the COMMON option of the SCALE operand. Specify SCALE(COMMON) if you want the same scale to be used to plot both variables.

For an explanation of the other VPLOT2 operands, refer to the descriptions of the VPLOT operands.

The following commands plot terminal response time with asterisks (*) and transaction use rate with plus signs (+):

```
VPLOT TERM RESP          CHAR1(*)
VPLOT2 TRANSACTION USE RATE CHAR1(+)
END
RUN
```


Variables

The variables you use with a report command determine what the report describes. If, for example, you use the command `GRAF TERMINAL USE` (where `TERMINAL USE` is the variable), then you get a report that graphs terminal use. This chapter describes how to create a full variable name, and provides details on all available variables.

You can specify the following two types of options with variables:

- **Variable options** – These options change either the type of data that a variable represents or its format. For example, the type of data can be changed from a count to a rate, or the format can be changed from one decimal place to three. Variable options are described in the chapter titled “Variable Options.”
- **ID options** – When used with variables, these options change the scope of the data the variables represent. For example, the scope of the data can be changed from transaction use for all transactions to the transaction use for transaction Cinq only. ID options are described in the chapter titled “ID Options.”

Structure of Variable Names

Report-writer variables have the following structure:

[CICS] [*record-type*] *variable-class* *variable-name*

Report-writer variables have the following parts:

Part	Description
CICS	Signals to the report writer that the variable is a Unicenter CA-Explore variable. This is the default.
<i>record-type</i>	<p>The type of record the variable's data comes from. You can specify one of the following:</p> <p>INTERval – Information on CICS resource usage. The collection of this data is based upon user-defined time intervals and involves low overhead.</p> <p>PERFormance – Information on task activity.</p> <p>SUMmary – Information from data created with the SUMMARIZE utility option.</p> <p>SYStem – Information from system threshold records.</p> <p>The default record type for all variables is PERFormance.</p> <p>The record type a variable represents affects which ID options can be used with it to limit the scope of a report.</p>
<i>variable-class</i>	One of the variable classes listed on the following page. For example, COMMAND and DATASET are both valid variable classes for the PERFormance record type.
<i>variable-name</i>	A variable name from the variable class selected. For example, as shown later in this chapter, both ERRORS and TIME are valid variable names for the variable class COMMAND.

Variable Descriptions

The following sections list and describe all report-writer variables. The variables are presented alphabetically according to record type. The following tables list the variable classes valid for each record type; the variable names for each class are described in subsequent sections.

Interval Record Type

Variable Class	Variable Class	Variable Class
DATASET	GETVIS	TRANsaction
DSA	REGION	

PERformance Record Type

Variable Class	Variable Class	Variable Class
COMMAND	PROGram	TERMinal
DATASET	RESOURCE	THRESHOLD
DLI	SECURITY	TOTAL
EXIT	SQL	TRANsaction
IRC	STORAGE	TRANSIENT
ISC	TABLE	WAIT
JOURNAL	TEMPSTOR	

REVIEW Record Type

Variable Class	Variable Class	Variable Class
DATASET	TERMinal	TRANsaction
DSA	THRESHOLD	

Summary Record Type

Variable Class
TERMinal
TRANsaction

System Record Type

Variable Class
THRESHOLD

Units of Measure

All variables that represent time are displayed in seconds.

INTErval Record Type for CICS

Variable class descriptions and the available ID options for the Interval Record type are discussed in this section.

DATASET Variable Class

You can specify FILE in place of DATASET for this variable class.

Record Type	Class	Variable Name	Description
INTErval	DATASET	BPR	Buffer performance rating (the ratio of the number of file requests to corresponding EXCPs)
INTErval	DATASET	CASPLITs	Number of CA splits
INTErval	DATASET	CISPLITs	Number of CI splits
INTErval	DATASET	DATA-BPR	Buffer performance rating (the ratio of the number of file requests to corresponding data component EXCPs)
INTErval	DATASET	DATA-EXCPs	Number of data EXCPs
INTErval	DATASET	DELETEDs	Number of records deleted
INTErval	DATASET	EXCPs	Number of EXCPs
INTErval	DATASET	EXTENDs	Number of dataset extends
INTErval	DATASET	INDX-BPR	Buffer performance rating (the ratio of the number of file requests to corresponding index component EXCPs)
INTErval	DATASET	INDX-EXCPs	Number of index EXCPs
INTErval	DATASET	INSERTs	Number of records inserted
INTErval	DATASET	LSRPOOL	LSR pool ID
INTErval	DATASET	REQuests	Number of file requests
INTErval	DATASET	RETRIEVEs	Number of records retrieved
INTErval	DATASET	UPDATEs	Number of records updated

ID Options

The following ID options can be used with any INTerval DATASET variable:

- CICSID
- ENDTIME
- FILEID
- JOBNAME
- LSRID
- STARTTIME
- VMID

DSA Variable Class (Variables for CICS 2.3 and TS 1.1)

Record Type	Class	Variable Name	Description
INTerval	DSA	%ALLOcated	<p>Percentage of DSA allocated.</p> <ul style="list-style-type: none"> ■ For CICS 2.3, this specifies its 1 DSA. ■ For CICS TS 1.1, this specifies its 8 DSAs.
INTerval	DSA	%AVAILable	<p>Percentage of DSA available (not used).</p> <ul style="list-style-type: none"> ■ For CICS 2.3, this specifies its 1 DSA. ■ For CICS TS 1.1, this specifies its 8 DSAs.
INTerval	DSA	ALLOcated	<p>Number of DSA allocated pages.</p> <ul style="list-style-type: none"> ■ For CICS 2.3, this specifies its 1 DSA. ■ For CICS TS 1.1, this specifies its 8 DSAs.

Record Type	Class	Variable Name	Description
INterval	DSA	AVAILable	<p>Number of DSA available (not used) pages.</p> <ul style="list-style-type: none"> ■ For CICS 2.3, this specifies its 1 DSA. ■ For CICS TS 1.1, this specifies its 8 DSAs.
INterval	DSA	COMPressions	<p>Number of DSA storage compressions.</p> <ul style="list-style-type: none"> ■ For CICS 2.3, this specifies its 1 DSA. ■ n/a to CICS TS 1.1
INterval	DSA	MAXBLOCK	<p>Average maximum block size of DSA available (not used) pages.</p> <ul style="list-style-type: none"> ■ For CICS 2.3, this specifies its 1 DSA. ■ For CICS TS 1.1, this specifies its 8 DSAs.
INterval	DSA	PAGEs	<p>Number of DSA pages.</p> <ul style="list-style-type: none"> ■ For CICS 2.3, this specifies its 1 DSA. ■ For CICS TS 1.1, this specifies its 8 DSAs.
INterval	DSA	SUBPOOL1	<ul style="list-style-type: none"> ■ Number of DSA pages allocated to subpool 1, the Control subpool, for CICS 2.3 ■ Number of pages allocated to the User DSA (UDSA), below the 16Mb line, for CICS TS 1.1.

Record Type	Class	Variable Name	Description
INTerval	DSA	SUBPOOL2	<ul style="list-style-type: none"> ■ Number of DSA pages allocated to subpool 2, the Teleprocessing subpool, for CICS 2.3 ■ Number of pages allocated to the CICS DSA (CDSA), below the 16Mb line, for CICS TS 1.1.
INTerval	DSA	SUBPOOL3	<p>Not used for CICS 2.3</p> <ul style="list-style-type: none"> ■ Number of pages allocated to the Shared DSA (SDSA), below the 16Mb line, for CICS TS 1.1.
INTerval	DSA	SUBPOOL4	<ul style="list-style-type: none"> ■ Number of DSA pages allocated to Subpool 4, the Task Subpool, for CICS 2.3 ■ Number of pages allocated to the Read-only DSA (RDSA), below the 16Mb line, for TS 1.1
INTerval	DSA	SUBPOOL5	<ul style="list-style-type: none"> ■ Number of DSA pages allocated to Subpool 5, the Shared Subpool, for CICS 2.3. ■ Number of pages allocated to the Extended User DSA (EUDSA), above the 16Mb line, for TS 1.1.
INTerval	DSA	SUBPOOL6	<ul style="list-style-type: none"> ■ Number of DSA pages allocated to Subpool 6, the VTAM-RPL Subpool, for CICS 2.3. ■ Number of pages allocated to the Extended CICS DSA (ECDSA), above the 16Mb line, for TS 1.1.

Record Type	Class	Variable Name	Description
INTerval	DSA	SUBPOOL7	<ul style="list-style-type: none"> ■ Not used for CICS 2.3. ■ Number of pages allocated to the Extended Shared DSA (ESDSA), above the 16Mb line.
INTerval	DSA	SUBPOOL8	<ul style="list-style-type: none"> ■ Number of DSA pages allocated to Subpool 8, the Program Subpool, for CICS 2.3. ■ Number of pages allocated to the Extended Read-only DSA (ERDSA), above the 16Mb line, for TS 1.1.

More About CICS TS 1.1 Dynamic Storage Areas

There are eight DSAs in CICS TS for VSE/ESA R1 – four above the 16MB line and four below the 16MB line. CICS allocates storage for them according to the values you specify in the new DSALIM and EDSALIM system initialization parameters. These values represent the upper limit of the total storage that CICS can allocate to the DSAs.

DSAs Below the 16MB Line:

- **UDSA** – The user DSA. The user-key storage area for all user-key task-lifetime storage below the 16MB line.
- **RDSA** – The read-only DSA. The key-0 storage area for all reentrant programs and tables below the 16MB line.
- **SDSA** – The shared DSA. The user-key storage area for any reentrant user-key RMODE(24) programs, and for any storage obtained by programs issuing EXEC CICS GETMAIN commands for storage below the 16MB line with the SHARED option.
- **CDSA** – The CICS DSA. The CICS-key storage area for all non-reentrant CICS-key RMODE(24) programs, all CICS-key task-lifetime storage below the 16MB line, and for all CICS control blocks that reside below the 16MB line.

DSAs Above the 16MB Line:

- **EUDSA** – The extended user DSA. The user-key storage area for all user-key task-lifetime storage above the 16MB line.
- **ERDSA** – The extended read-only DSA. The key-0 storage area for all reentrant programs and tables above the 16MB line.
- **ESDSA** – The extended shared DSA. The user-key storage area for any reentrant user-key RMODE(ANY) programs. For any storage obtained by programs issuing EXEC CICS GETMAIN commands for storage above the 16MB line with the SHARED option.
- **ECDSA** – The extended CICS DSA. The CICS-key storage area for all non-reentrant CICS-key RMODE(ANY) programs, all CICS-key task-lifetime storage above the 16MB line, and for all CICS control blocks that reside above the 16MB line.

ID Options

The following ID options can be used with any INterval DSA variable:

- CICSID
- ENDTIME
- JOBNAME
- STARTTIME
- VMID

GETVIS Variable Class

Record Type	Class	Variable Name	Description
INterval	GETVIS	%AVAILABLE	Average percentage of GETVIS available (not used)
INterval	GETVIS	%USED	Average percentage of GETVIS used
INterval	GETVIS	31-%AVAILABLE	Average percentage of 31-bit GETVIS available (not used)
INterval	GETVIS	31-%USED	Average percentage of 31-bit GETVIS used

INTERVAL	GETVIS	31-AVAILABLE	Average 31-bit GETVIS available (not used), in kilobytes
INTERVAL	GETVIS	31-MAXBLOCK	Average maximum block size of available 31-bit GETVIS, in kilobytes
INTERVAL	GETVIS	31-SIZE	Partition 31-bit GETVIS size, in kilobytes
INTERVAL	GETVIS	31-USED	Average 31-bit GETVIS used, in kilobytes
INTERVAL	GETVIS	AVAILABLE	Average GETVIS available (not used), in kilobytes
INTERVAL	GETVIS	MAXBLOCK	Average maximum block size of available GETVIS, in kilobytes
INTERVAL	GETVIS	SIZE	Partition GETVIS size, in kilobytes
INTERVAL	GETVIS	USED	Average GETVIS used, in kilobytes

ID Options

The following ID options can be used with any INTERVAL GETVIS variable:

- CICSID
- ENDTIME
- JOBNAME
- STARTTIME
- VMID

REGION Variable Class

Record Type	Class	Variable Name	Description
INTerval	REGION	CPUtime	Total CPU time consumed by the CICS region
INTerval	REGION	SIOs	Number of SIOs executed by the CICS region
INTerval	REGION	THRESHOLDs	Number of thresholds triggered in the CICS region

ID Options

The following ID options can be used with any INTerval REGION variable:

- CICSID
- ENDTIME
- JOBNAME
- STARTTIME
- VMID

TRANsaction Variable Class

Record Type	Class	Variable Name	Description
INTerval	TRANsaction	CPUtime	Average CPU time, in seconds
INTerval	TRANsaction	DLItime	Average time, in seconds, spent serving DL/I requests
INTerval	TRANsaction	EXPCtime	Average time, in seconds, spent in Unicenter CA-Explore code
INTerval	TRANsaction	I/Otime	Average time, in seconds, spent waiting for I/O
INTerval	TRANsaction	LIFETIME	Average transaction lifetime, in seconds
INTerval	TRANsaction	PGMtime	Average time, in seconds, spent in program code, including both user and CICS code
INTerval	TRANsaction	RESOURCEtime RSCStime	Average time, in seconds, spent waiting for any resource
INTerval	TRANsaction	STGSUSPtime	Average time, in seconds, spent suspended due to storage
INTerval	TRANsaction	SUSPtime	Average time, in seconds, spent suspended
INTerval	TRANsaction	TCPUtime	Total CPU time for the transaction including user, CICS, and Unicenter CA-Explore time

Record Type	Class	Variable Name	Description
INTerval	TRANsaction	TMIOtime	Average time, in seconds, spent waiting for terminal
INTerval	TRANsaction	TRACetime CMPtime	Average time spent, in seconds, in trace code
INTerval	TRANsaction	USE	Number of transactions executed
INTerval	TRANsaction	WAITtime	Average time, in seconds, spent waiting for an event control block
INTerval	TRANsaction	WTRtime	Average time, in seconds, spent waiting to run

ID Options

The following ID options can be used with any INTerval TRANsaction variable:

- CICSID
- ENDTIME
- JOBNAME
- STARTTIME
- VMID

PERformance Record Type

Variable class descriptions and the available ID options for the Performance Record Type discussed in this section.

COMMAND Variable Class

Record Type	Class	Variable Name	Description
PERformance	COMMAND	ERRORS	Number of command-level request errors
PERformance	COMMAND	MAXRESPonse	Maximum response time, in seconds, of any command-level request
PERformance	COMMAND	REQUESTs	Number of command-level requests per transaction
PERformance	COMMAND	RESPonse	Average response time, in seconds, per command-level request
PERformance	COMMAND	TIME	Time, in seconds, spent in command-level interface per transaction

ID Options

The following ID options can be used with any COMMAND variable:

- ABCODE
- APPLID
- CICSID
- EPID
- ENDTIME
- JOBNAME, NETNAME
- OPERID
- OPERNAME
- STARTTIME
- TERMID
- TRANID, TRANNUM
- USERID
- UTRANID
- VMID
- VTAMLU

DATASET Variable Class

You can specify FILE in place of DATASET for this variable class.

Record Type	Class	Variable Name	Description
PERformance	DATASET	ADDs	Number of file adds
PERformance	DATASET	BROWSEs	Number of file browses
PERformance	DATASET	BUFFWAITime	Average buffer wait time, in seconds
PERformance	DATASET	CASPLITs	Number of CA splits
PERformance	DATASET	CISPLITs	Number of CI splits
PERformance	DATASET	CI/CASPLITS CI/CA SPLITS	Number of CI and CA split waits
PERformance	DATASET	DATATABLE-REQuests	Number of data table requests
PERformance	DATASET	DELetes	Number of file deletes
PERformance	DATASET	MAXI/Otime	Maximum file I/O time, in seconds
PERformance	DATASET	MAXREQtime	Maximum file request service time, in seconds
PERformance	DATASET	PSTRINGWAITime	Average pseudo-string wait time, in seconds
PERformance	DATASET	READNUpdts GETNU	Number of file reads-no-update
PERformance	DATASET	READs	Number of read-updates, read-no-updates, and browses
PERformance	DATASET	READUpdts GETU	Number of file reads for update

Record Type	Class	Variable Name	Description
PERformance	DATASET	REQtime	Average file request service time, in seconds
PERformance	DATASET	SERVTIME	Average file I/O service time, in seconds
PERformance	DATASET	SPLITCNT	Number of CI/CA splits
PERformance	DATASET	STRINGWAITime	Average string wait time, in seconds
PERformance	DATASET	TYPE1	File type 1 description: ? Unknown B BDAM D Data table E DATACOM or SUPRA R Remote T TOTAL database U User file V VSAM
PERformance	DATASET	TYPE2	File type 2 description: ? Unknown B BDAM keyed E ESDS K KSDS L LDS N BDAM non-keyed R RSDS S System data table T TOTAL database U User data table

Record Type	Class	Variable Name	Description
PERformance	DATASET	UPDATEs	Number of file updates
PERformance	DATASET	USE USAGE RW REQUESTs	Number of file requests
PERformance	DATASET	VSAMPSOS PSOS	Number of pseudo short-on-string waits
PERformance	DATASET	VSAMSOS SOS	Number of short-on-string waits
PERformance	DATASET	VSAMSOSS SOSS	Number of short-on-shared-string waits
PERformance	DATASET	VSAMWBUF WBUF	Number of buffer waits
PERformance	DATASET	VSAMXCL XCL	Number of exclusive control waits
PERformance	DATASET	WAITS	Number of file waits
PERformance	DATASET	WRITEs	Number of updates, adds, and deletes

ID Options

The following ID options can be used with PERFORMANCE DATASET variables:

- ABCODE
- APPLID
- CICSID
- FILEID
- FTYPE1
- FTYPE2
- JOBNAME
- OPERID
- OPERNAME
- PROGID
- STARTTIME
- TRANID
- USERID
- UTRANID
- VOLSER
- TRANNUM
- VTAMLU

DLI Variable Class

Record Type	Class	Variable Name	Description
PERformance	DLI	BUFPERF	DL/I buffer performance rating (requests divided by SIOs)
PERformance	DLI	CHKP	Number of DL/I checkpoint requests Note: This value is collected only by Unicenter CA-Explore for CICS Release 7.x. It is not available in data collected by Unicenter CA-Explore for CICS Release 6.x.
PERformance	DLI	DELETES	Number of DL/I delete requests
PERformance	DLI	GETNEXT GN	Number of DL/I GET next requests
PERformance	DLI	GETUNIQUE GU	Number of DL/I GET unique requests
PERformance	DLI	GHN	Number of DL/I GET hold next requests
PERformance	DLI	GHNP	Number of DL/I GET hold next within parent requests
PERformance	DLI	GHU	Number of DL/I GET hold unique requests
PERformance	DLI	GNP	Number of DL/I GET next within parent requests
PERformance	DLI	INSERTs	Number of DL/I insert requests
PERformance	DLI	MAXI/OTime	Maximum DL/I file SIO time, in seconds
PERformance	DLI	MAXREQtime	Maximum DL/I file request service time, in seconds
PERformance	DLI	READs	Number of DL/I read requests, including GETNEXT, GNP, GU, GHU, GHN, and GHNP

Record Type	Class	Variable Name	Description
PERformance	DLI	REPLACEs	Number of DL/I replace requests
PERformance	DLI	REQTIME	Average DL/I file request service time, in seconds
PERformance	DLI	SERVTIME	Average DL/I file request service time, in seconds
PERformance	DLI	SIOs	Number of DL/I file SIOs
PERformance	DLI	SIOTime	Average DL/I file SIO time, in seconds
PERformance	DLI	TYPE1	File type 1 description: D DL/I R Remote ? Unknown
PERformance	DLI	TYPE2	File type 2 description: A DL/I ACB P DL/I PSB D DL/I DBD F DL/I function M Module or program ? Unknown
PERformance	DLI	USE USAGE REQUESTs IO	Number of DL/I file requests
PERformance	DLI	WRITEs	Number of DL/I write requests, including replaces, deletes, and inserts

Special Considerations

To avoid data duplication when you create reports using PERFORMANCE DLI variables, you should do one of the following:

Specify the ID option DLITYPE1 with the primary report command to organize the report by file type 1.

Specify the ID option DLITYPE2 and an identifier with the primary report command or with each variable to restrict the report to data of a single file type 2.

The following commands show correct ways of running reports using DLI variables:

```
TAB DLITYPE2 DLI SIO AND DLI SIOTIME  
END
```

```
TAB USERID DLITYPE2(D) DLI SIO AND DLI SIOTIME  
END
```

```
TAB DLI SIO DLITYPE2(A) AND DLI SIOTIME DLITYPE2(A)  
END
```

ID Options

The following ID options can be used with PERformance DLI variables:

- ABCODE
- APPLID
- CICSID
- DLITYPE1
- DLITYPE2
- ENDTTIME
- FILEID
- JOBNAME
- NETNAME
- OPERID
- OPERNAME
- STARTTIME
- TERMID
- TRANID
- TRANNUM
- USERID
- VMID
- VTAMLU

In addition, the ID option DLIPPOOL can be used with PERformance DLI variables when the report is restricted to data with DLITYPE2 of D. For example, you can use the following command:

```
TAB DLIPPOOL DLITYPE2(D) DLI SIOs
```

For reports on data with DLITYPE2 of A or D, only the following PERformance DLI variables are valid:

- DLI SIOs
- DLI SIOTIME
- DLI MAXI/OTime

EXIT (Global User Exits) Variable Class

Record Type	Class	Variable Name	Description
PERformance	EXIT	COUNT	Number of exit calls per transaction
PERformance	EXIT	MAXRESPonse	Maximum response time, in seconds, from user exit
PERformance	EXIT	RESPonse	Average response time, in seconds, per user exit call
PERformance	EXIT	TIME	Time, in seconds, spent in user exit per transaction

ID Options

The following ID options can be used with any EXIT variable:

- ABCODE
- APPLID
- CICSID
- ENDTIME
- EXITID
- JOBNAME
- NETNAME
- OPERID
- OPERNAME
- STARTTIME
- TERMID
- TRANID
- UTRANID
- TRANNUM
- USERID
- VMID
- VTAMLU

IRC (Inter-Region Communication) Variable Class

Record Type	Class	Variable Name	Description
PERformance	IRC	MAXRESPonse	Maximum response time, in seconds, of any IRC request
PERformance	IRC	READS	Number of inter-region communication read requests per transaction
PERformance	IRC	REQUESTS	Number of inter-region communication requests per transaction
PERformance	IRC	RESPonse	Average response time, in seconds, per IRC request
PERformance	IRC	TIME	Time, in seconds, spent waiting for inter-region communication per transaction
PERformance	IRC	WRITES	Number of inter-region communication writes per transaction

ID Options

The following ID options can be used with any IRC variable:

- ABCODE
- APPLID
- CICSID
- ENDTIME
- JOBNAME
- NETNAME
- OPERID
- OPERNAME
- STARTTIME
- SYSIDNT
- TERMID
- TRANID
- TRANNUM
- USERID
- UTRANID
- VMID
- VTAMLU

ISC (Intersystem Communication) Variable Class

Record Type	Class	Variable Name	Description
PERformance	ISC	MAXRESPonse	Maximum response time, in seconds, of any ISC request
PERformance	ISC	REQUESTs	Number of intersystem communication requests per transaction
PERformance	ISC	RESPonse	Average response time, in seconds, per ISC request
PERformance	ISC	TIME	Time, in seconds, spent waiting for intersystem communication per transaction

ID Options

The following ID options can be used with any ISC variable:

- ABCODE
- APPLID
- CICSID
- ENDTIME
- JOBNAME
- NETNAME
- OPERID
- OPERNAME
- STARTTIME
- SYSIDNT
- TERMID
- TRANID
- TRANNUM
- USERID
- UTRANID
- VMID
- VTAMLU

JOURNAL Variable Class

Record Type	Class	Variable Name	Description
PERformance	JOURNAL	MAXRESPonse	Maximum response time, in seconds, of any journal request
PERformance	JOURNAL	REQUESTs	Number of journal requests per transaction
PERformance	JOURNAL	RESPonse	Average response time, in seconds, per journal request
PERformance	JOURNAL	TIME	Time, in seconds, spent waiting in journal control per transaction

ID Options

The following ID options can be used with any JOURNAL variable:

- ABCODE
- APPLID
- CICSID
- ENDTIME
- JOBNAME
- NETNAME
- OPERID
- OPERNAME
- STARTTIME
- SYSIDNT
- TERMID
- TRANID
- TRANNUM
- USERID
- UTRANID
- VMID
- VTAMLU

PROGram Variable Class

Record Type	Class	Variable Name	Description
PERformance	PROGram	ABENDs	Number of program abends
PERformance	PROGram	FETCHs	Number of program fetches
PERformance	PROGram	LINKs	Number of program links
PERformance	PROGram	LOADs	Number of program loads
PERformance	PROGram	MAXRESPonsetime	Maximum program response time, in seconds
PERformance	PROGram	RESPonsetime	Average program response time, in seconds
PERformance	PROGram	USE	Program use count
PERformance	PROGram	XCTLs	Number of program XCTLs

ID Options

The following ID options can be used with PERformance PROGram variables:

- ABCODE
- APPLID
- CICSID
- ENDTIME
- JOBNAME
- NETNAME
- OPERID
- OPERNAME
- STARTTIME
- SYSIDNT
- TERMID
- TRANID
- TRANNUM
- USERID
- UTRANID
- VMID
- VTAMLU

RESOURCE Variable Class

Record Type	Class	Variable Name	Description
PERformance	RESOURCE	MAXRESPonse	Maximum response time, in seconds, for any resource request
PERformance	RESOURCE	REQUESTs	Number of resource requests per transaction
PERformance	RESOURCE	RESPonse	Average response time, in seconds, per resource request
PERformance	RESOURCE	TIME	Time, in seconds, spent in resource control per transaction

ID Options

The following ID options can be used with any RESOURCE variable:

- ABCODE
- APPLID
- CICSID
- ENDTIME
- JOBNAME
- NETNAME
- OPERID
- OPERNAME
- RSCSID
- STARTTIME
- TERMID
- TRANID
- TRANNUM
- USERID
- UTRANID
- VMID
- VTAMLU

SECURITY Variable Class

Record Type	Class	Variable Name	Description
PERformance	SECURITY	MAXRESPonse	Maximum response time, in seconds, of any external security call
PERformance	SECURITY	REQUESTs	Number of external security calls per transaction
PERformance	SECURITY	RESPonse	Average response time, in seconds, per external security call
PERformance	SECURITY	TIME	Time, in seconds, spent in external security manager per transaction

ID Options

The following ID options can be used with any SECURITY variable:

- ABCODE
- APPLID
- CICSID
- ENDTIME
- JOBNAME
- NETNAME
- OPERID
- OPERNAME
- RSCSID
- STARTTIME
- TERMID
- TRANID
- TRANNUM
- USERID
- UTRANID
- VMID
- VTAMLU

SQL Variable Class

System	Record Type	Class	Variable Name	Description
CICS	PERformance	SQL	MAXRESPonse	Maximum response time, in seconds, of any SQL/DS request
CICS	PERformance	SQL	REQUESTs	Number of SQL/DS requests per transaction
CICS	PERformance	SQL	RESPonse	Average response time, in seconds, per SQL/DS request

ID Options

The following ID options can be used with PERformance SQL variables:

- NETNAME
- SQLID
- TRANNUM
- APPLID
- NETNAME
- SQLTYPE
- USERID
- CICSID
- OPERID
- STARTTIME
- UTRANID
- ENDTIME
- OPERNAME
- TERMID
- VMID
- FILEID
- TRAMID
- VTAMLU

STORAGE Variable Class

Record Type	Class	Variable Name	Description
PERformance	STORAGE	AMOUNT	Average storage used per transaction
PERformance	STORAGE	REQUESTS	Average GETMAIN count per transaction
PERformance	STORAGE	SIZE	Average GETMAIN request size (in bytes)

ID Options

The following ID options can be used with PERformance STORAGE variables:

- ABCODE
- APPLID
- CICSID
- ENDTIME
- JOBNAME
- NETNAME
- OPERID
- OPERNAME
- STARTTIME
- STORID
- TERMID
- TRANID
- TRANNUM
- USERID
- UTRANID
- VMID
- VTAMLU

TABLE Variable Class

Record Type	Class	Variable Name	Description
PERformance	TABLE	MAXRESPonse	Maximum response time, in seconds, of any table manager program
PERformance	TABLE	REQUESTs	Number of table manager programs per transaction
PERformance	TABLE	RESPonse	Average response time, in seconds, per table manager program
PERformance	TABLE	TIME	Time, in seconds, spent in external security manager per transaction

ID Options

The following ID options can be used with any TABLE variable:

- ABCODE
- APPLID
- CICSID
- ENDTIME
- JOBNAME
- NETNAME
- OPERID
- OPERNAME
- RSCSID
- STARTTIME
- TERMID
- TRANID
- TRANNUM
- USERID
- UTRANID
- VMID
- VTAMLU

TEMPSTOR Variable Class

Record Type	Class	Variable Name	Description
PERformance	TEMPSTOR	BGET	Average number of temporary storage bytes per GET
PERformance	TEMPSTOR	BPUT	Average number of temporary storage bytes per PUT
PERformance	TEMPSTOR	BREQ	Average number of temporary storage bytes per request
PERformance	TEMPSTOR	I/OTIME	Average temporary storage I/O wait time, in seconds
PERformance	TEMPSTOR	MAXRESP	Maximum temporary storage request time, in seconds
PERformance	TEMPSTOR	PUTs	Number of temporary storage PUT requests
PERformance	TEMPSTOR	PUTSAUX	Number of temporary storage PUTs to auxiliary storage
PERformance	TEMPSTOR	PUTSMAIN	Number of temporary storage PUTs to main storage
PERformance	TEMPSTOR	READs GETs	Number of temporary storage GET requests
PERformance	TEMPSTOR	REQUESTs	Number of temporary storage requests
PERformance	TEMPSTOR	RESPONSE	Average temporary storage request time, in seconds, per request
PERformance	TEMPSTOR	SUSPENDs	Number of temporary storage suspends
PERformance	TEMPSTOR	SUSPTIME	Average temporary storage suspend time, in seconds

ID Options

The following ID options can be used with PERformance TEMPSTOR variables:

- ABCODE
- APPLID
- CICSID
- ENDTIME
- JOBNAME
- NETNAME
- OPERID
- OPERNAME
- STARTTIME
- TERMID
- TRANNUM
- TSTGID
- USERID
- UTRANID
- VMID
- VTAMLU

TERMinal Variable Class

Record Type	Class	Variable Name	Description
PERformance	TERMinal	INPUTs READs	Number of terminal input messages
PERformance	TERMinal	INPUTSIZE	Average input message size
PERformance	TERMinal	I/OTIME	Time, in seconds, spent doing reads and writes per transaction
PERformance	TERMinal	MAXRESP	Maximum terminal response time, in seconds
PERformance	TERMinal	MESSAGES	Number of messages
PERformance	TERMinal	MSGSIZE	Average message size in bytes
PERformance	TERMinal	OUTPUTS USE USAGE WRITES	Number of terminal output messages
PERformance	TERMinal	OUTPUTSIZE	Average output message size
PERformance	TERMinal	RESPonsetime	Average terminal response time, in seconds

ID Options

Any ID option except LSRID and VOLSER can be used with PERFORMANCE TERMinal variables.

THRESHOLD Variable Class

Record Type	Class	Variable Name	Description
PERformance	THRESHOLD	COUNT	Number of thresholds exceeded
PERformance	THRESHOLD	LIMIT	Threshold limit value
PERformance	THRESHOLD	MAXVALUE	Maximum value that exceeded the limit
PERformance	THRESHOLD	MINVALUE	Minimum value that exceeded the limit
PERformance	THRESHOLD	VALUE	Average value that exceeded the limit

ID Options

The following ID options can be used with any PERformance THRESHOLD variable:

- ABCODE
- APPLID
- CICSID
- ENDTIME
- HRCLAS
- JOBNAME
- NETNAME
- OPERID
- OPERNAME
- STARTTIME
- TERMID
- THRSID
- THRTYPE
- TRAND
- TRANNUM
- TRSCEID
- USERID
- UTRANID
- VMID
- VTAMLU

TOTAL (TOTAL Database) Variable Class

Record Type	Class	Variable Name	Description
PERformance	TOTAL	ADDs	Number of TOTAL database add requests
PERformance	TOTAL	BROWSEs	Number of TOTAL database browse requests
PERformance	TOTAL	DELETEs	Number of TOTAL database delete requests
PERformance	TOTAL	READs	Number of TOTAL database read requests
PERformance	TOTAL	READSUM	Number of TOTAL database reads and browses
PERformance	TOTAL	USE IO REQUESTS	Number of TOTAL database file requests
PERformance	TOTAL	WRITEs	Number of TOTAL database write requests
PERformance	TOTAL	WRITESUM	Number of TOTAL database adds, deletes, and writes

ID Options

The following ID options can be used with any TOTAL variable:

- ABCODE
- APPLID
- CICSID
- ENDTIME
- JOBNAM
- NETNAME
- OPERID
- OPERNAME
- STARTTIME
- TERM
- TRANID
- TRANNUM
- UTRANID
- VMID
- VTAMLU

TRANsaction Variable Class

Record Type	Class	Variable Name	Description
PERformance	TRANsaction	ABCODE	Transaction abend code
PERformance	TRANsaction	ABENDS	Number of abends
PERformance	TRANsaction	AVGETMains	Average GETMAIN count per transaction
PERformance	TRANsaction	BMSWAITs	Average number of BMS waits
PERformance	TRANsaction	CICSWaitS	Average number of CICS waits
PERformance	TRANsaction	COST	Total cost per report line. This is the same as the variable REPORT COST TOTAL, which is described with the COST command.
PERformance	TRANsaction	CPUtime	Average CPU time, in seconds
PERformance	TRANsaction	DCOUNT	Average number of transient datasets used
PERformance	TRANsaction	DCPWAITs	Average number of waits for the destination control program
PERformance	TRANsaction	DEQUEUEs	Average number of dequeues
PERformance	TRANsaction	DISPATCHs	Average number of dispatches
PERformance	TRANsaction	DLItime	Average time, in seconds, spent serving DL/I requests
PERformance	TRANsaction	DLIWAITs	Average number of DL/I waits
PERformance	TRANsaction	EIPWAITs	Average number of command-level waits
PERformance	TRANsaction	ENDDATE	Transaction end date
PERformance	TRANsaction	ENDTIME	Transaction end time

Record Type	Class	Variable Name	Description
PERformance	TRANsaction	ENQUEUEs	Average number of enqueues
PERformance	TRANsaction	EXPCtime	Average time, in seconds, spent in Unicenter CA-Explore code
PERformance	TRANsaction	FCOUNT	Average number of files accessed
PERformance	TRANsaction	FCPWAITs	Average number of waits for the file control program
PERformance	TRANsaction	FREEM	Average FREEMAIN count per transaction
PERformance	TRANsaction	GETMAIN	Average GETMAN count
PERformance	TRANsaction	GETMSIZE	Average GETMAIN request size (in bytes)
PERformance	TRANsaction	ICPWAITs	Average number of waits for the interval control program
PERformance	TRANsaction	I/Otime	Average time, in seconds, spent waiting for I/O
PERformance	TRANsaction	JCPWAITs	Average number of waits for the journal control program
PERformance	TRANsaction	LIFetime	Average transaction lifetime, in seconds
PERformance	TRANsaction	MAXCPUtime	Maximum CPU time, in seconds
PERformance	TRANsaction	MAXEXPCtime	Maximum time, in seconds, spent in Unicenter CA-Explore code
PERformance	TRANsaction	MAXDLItime	Maximum DL/I time, in seconds
PERformance	TRANsaction	MAXGETMreqsize	Maximum GETMAIN request size (in bytes)

Record Type	Class	Variable Name	Description
PERformance	TRANsaction	MAXI/Otime	Maximum time, in seconds, spent waiting for I/O
PERformance	TRANsaction	MAXLIFE	Maximum transaction lifetime, in seconds
PERformance	TRANsaction	MAXPGMtime	Maximum time, in seconds, spent in program code, including both user and CICS code
PERformance	TRANsaction	MAXRESourcetime MAXRSCS	Maximum time, in seconds, spent waiting for a resource
PERformance	TRANsaction	MAXSTGSUSPtime	Maximum time, in seconds, spent suspended due to DSA storage
PERformance	TRANsaction	MAXTRACetime	Maximum time, in seconds, spent in trace code
PERformance	TRANsaction	MAXSUSPtime	Maximum time, in seconds, spent suspended
PERformance	TRANsaction	MAXTMIOtime	Maximum time, in seconds, spent waiting for terminal
PERformance	TRANsaction	MAXWAITtime	Maximum time, in seconds, spent waiting for an event control block
PERformance	TRANsaction	MAXWTRtime	Maximum time, in seconds, spent waiting to run
PERformance	TRANsaction	MROWAITs	Average number of multiple region option waits
PERformance	TRANsaction	MULTWaits	Average number of multiple waits for resources of any type

Record Type	Class	Variable Name	Description
PERformance	TRANsaction	PCOUNT	Average number of programs used
PERformance	TRANsaction	PCPWAITs	Average number of waits on the program control program
PERformance	TRANsaction	PGMtime	Average time, in seconds, spent in program code, including both user and CICS code
PERformance	TRANsaction	RESOURCEtime RSCS	Average time, in seconds, spent waiting for any resource
PERformance	TRANsaction	RTCLCKC	Transaction and terminal information. Value is always 1.
PERformance	TRANsaction	RTCMDLC	Average number of unique command-level calls per transaction
PERformance	TRANsaction	RTCNTRC	Transaction and terminal information. Value is always 1.
PERformance	TRANsaction	RTDBPIC	Average number of unique SQL/DS agents used per transaction
PERformance	TRANsaction	RTDLISC	Average number of unique DL/I files used per transaction
PERformance	TRANsaction	RTERMGC	Average number of unique exit resources used per transaction
PERformance	TRANsaction	RTEWANC	Average number of unique CICS subsystem wait types per transaction
PERformance	TRANsaction	RTFILEC	Average number of unique files used per transaction

Record Type	Class	Variable Name	Description
PERformance	TRANsaction	RTISCPC	Average number of unique ISC regions communicated with per transaction
PERformance	TRANsaction	RTJRNL	Average number of unique journal datasets used per transaction
PERformance	TRANsaction	RTMAIN	Transaction and terminal information. Value is always 1.
PERformance	TRANsaction	RTMROCC	Average number of unique IRC regions communicated with per transaction
PERformance	TRANsaction	RTPROGC	Average number of unique programs used per transaction
PERformance	TRANsaction	RTRHDRC	Transaction and terminal information. Value is always 1.
PERformance	TRANsaction	RTSTGTC	Average number of unique storage types acquired per transaction
PERformance	TRANsaction	RTDATC	Average number of unique transient datasets used per transaction
PERformance	TRANsaction	RTTHSPC	Average number of unique thresholds triggered per transaction
PERformance	TRANsaction	RTTMGPC	Average number of table manager requests per transaction

Record Type	Class	Variable Name	Description
PERformance	TRANsaction	RTTSTGC	Average number of unique temporary storage queues used per transaction
PERformance	TRANsaction	RTUEHDC	Average number of unique global user exits used per transaction
PERformance	TRANsaction	RTXSMGC	Average number of unique external security calls per transaction
PERformance	TRANsaction	SCPWAITs	Average number of waits for the storage control program
PERformance	TRANsaction	SINGLEWaits	Average number of single waits
PERformance	TRANsaction	SOSWs	Average number of short-on-storage waits
PERformance	TRANsaction	STARTDATE	Transaction start date
PERformance	TRANsaction	STARTTIME	Transaction start time
PERformance	TRANsaction	STGSUSPtime	Average time, in seconds, spent suspended due to storage
PERformance	TRANsaction	STORAge	Average DSA storage use per transaction
PERformance	TRANsaction	STORVIOLations	Number of storage violations
PERformance	TRANsaction	SUSPENDs	Average number of suspends
PERformance	TRANsaction	SUSPtime	Average time, in seconds, spent suspended

Record Type	Class	Variable Name	Description
PERformance	TRANsaction	SYSWAITs	Average number of system waits
PERformance	TRANsaction	TCPUtime	Total CPU time for the transaction including user, CICS, and Unicenter CA-Explore time
PERformance	TRANsaction	TCPWAITs	Average number of waits for the trace control program
PERformance	TRANsaction	TDPWAITs	Average number of waits on the transient data program
PERformance	TRANsaction	TMIOtime	Average time, in seconds, spent waiting for terminal
PERformance	TRANsaction	TRACetime	Average time, in seconds, spent in trace code
PERformance	TRANsaction	TRPWAITs	Average number of trace program waits
PERformance	TRANsaction	TSPWAITs	Average number of waits for the temporary storage program
PERformance	TRANsaction	USE USAGE	Number of transactions executed
PERformance	TRANsaction	WAITtime	Average time, in seconds, spent waiting for an event control block
PERformance	TRANsaction	WAITS	Average number of waits of any type
PERformance	TRANsaction	WORKSET WS	Average current storage usage (in bytes)
PERformance	TRANsaction	WTRtime	Average time, in seconds, spent waiting to run

Variables No Longer Valid

The following TRANsaction variables are not valid in version 5.73 and higher of Unicenter CA-Explore for CICS; however, they are included as part of another variable, as indicated in the following table:

This Variable Is No Longer Valid	But Is Included in the Value of This Variable
TRANsaction CICStime	TRANsaction PGMTIME
TRANsaction MAXCICStime	TRANsaction MAXPGMTIME
TRANsaction MAXUSERtime	TRANsaction MAXPGMTIME
TRANsaction USERtime	TRANsaction PGMTIME

Long, Medium, and Short Transactions

Some TRANsaction variables apply only to long, medium, or short transactions. The parameters that define each of these categories are set by the configuration override member for the CICS/VS region.

ID Options

All ID options except LSRID and VOLSER can be used with PERformance TRANsaction variables.

TRANSIENT Variable Class

Record Type	Class	Variable Name	Description
PERformance	TRANSIENT	I/OTIME	Average time, in seconds, spent waiting for I/O
PERformance	TRANSIENT	READs GETs INPUTs	Number of reads of transient datasets
PERformance	TRANSIENT	REQUESTs	Number of I/Os to transient datasets
PERformance	TRANSIENT	WRITEs PUTs OUTPUTs	Number of writes to transient datasets

ID Options

The following ID options can be used with PERformance TRANSIENT variables:

- ABCODE
- APPLID
- CICSID
- DCTID
- ENDTIME
- NETNAME
- OPERID
- OPERNAME
- STARTTIME
- TERMID
- TRANID
- TRANNUM
- USERID
- UTRANID
- VMID
- VTAMLU

WAIT Variable Class

Record Type	Class	Variable Name	Description
PERformance	WAIT	COUNT	Number of waits per transaction
PERformance	WAIT	MAXRESPonse	Maximum time, in seconds, of any wait
PERformance	WAIT	RESPonse	Average time, in seconds, per wait
PERformance	WAIT	TIME	Time, in seconds, spent waiting per transaction

ID Options

The following ID options can be used with any WAIT variable:

- ABCODE
- APPLID
- CICSID
- ENDTIME
- JOBNAME
- NETNAME
- OPERID
- OPERNAME
- STARTTIME
- TERMID
- TRANID
- TRANSNUM
- USERID
- UTRANID
- VMID
- VTAMLU
- WAITID

REVIEW Record Type

Variable class descriptions and the available ID options for the Review Record Type are discussed in this section.

DATASET Variable Class

Record Type	Class	Variable Name	Description
REVIEW	DATASET	REQuests	Number of file requests
REVIEW	DATASET	SERVtime	Average file I/O service time, in seconds
REVIEW	DATASET	SPLITs	Number of CI and CA splits

ID Options

The following ID options can be used with any REVIEW DATASET variable:

- CICSID
- JOBID (For Unicenter CA-Explore before Version 6.5, the jobname prints the CICS region ID.)

DSA Variable Class

Record Type	Class	Variable Name	Description
REVIEW	DSA	COMPressions	Number of DSA storage compressions. This is valid only for CICS 2.3. Compression values are not collected for CICS TS. For details, see the Unicenter Explore for CICS <i>Online User Guide</i> .
REVIEW	DSA	SOS	Number of short on storage occurrences

ID Options

The following ID options can be used with any REVIEW DSA variable:

- CICSID
- JOBID (For Unicenter CA-Explore before Version 6.5, the jobname prints the CICS region ID.)

TERMinal Variable Class

Record Type	Class	Variable Name	Description
REVIEW	TERMinal	RESPonsetime	Average response time, in seconds

ID Options

The following ID options can be used with any REVIEW TERMinal variable:

- CICSID
- JOBID (For Unicenter CA-Explore before Version 6.5, the jobname prints the CICS region ID.)

THRESHOLD Variable Class

Record Type	Class	Variable Name	Description
REVIEW	THRESHOLD	COUNT	Number of thresholds triggered

ID Options

The following ID options can be used with any REVIEW THRESHOLD variable:

- CICSID
- JOBID

TRANsaction Variable Class

Record Type	Class	Variable Name	Description
REVIEW	TRANsaction	ABENDs	Number of abends
REVIEW	TRANsaction	CPUtime	Average CPU time, in seconds
REVIEW	TRANsaction	LIFETIME	Average lifetime, in seconds
REVIEW	TRANsaction	TCPUtime	Total CPU time, in seconds
REVIEW	TRANsaction	USE	Number of transactions executed

ID Options

The following ID options can be used with any REVIEW TRANsaction variable:

- CICSID
- JOBID

SUMmary Record Type

Variable class descriptions and the available ID options for the Summary Record Type are discussed in this section.

TERMinal Variable Class

Record Type	Class	Variable Name	Description
SUMmary	TERMinal	INPUTs	Number of terminal input messages
SUMmary	TERMinal	INPUTSIZE	Average input message size, in bytes
SUMmary	TERMinal	MESSAGES	Total number of input and output messages
SUMmary	TERMinal	MSGSIZE	Average message size, in bytes
SUMmary	TERMinal	OUTPUTs	Number of terminal output messages
SUMmary	TERMinal	OUTPUTSIZE	Average output message size, in bytes
SUMmary	TERMinal	RESPonsetime	Average terminal response time, in seconds

ID Options

The following ID options can be used with any SUMmary TERMinal variable:

- APPLID
- CICSID
- ENDTIME
- JOBNAME
- OPERID
- STARTTIME
- TERMID
- TRANID
- USERID
- VMID
- VTAMLU

TRANsaction Variable Class

Record Type	Class	Variable Name	Description
SUMmary	TRANsaction	ABENDs	Number of abends
SUMmary	TRANsaction	CPUtime	Average CPU time, in seconds
SUMmary	TRANsaction	DLI-REQuests	Number of DL/I file requests
SUMmary	TRANsaction	DLItime	Average time spent serving DL/I requests
SUMmary	TRANsaction	EXPCtime	Average time spent in Unicenter CA-Explore code
SUMmary	TRANsaction	FC-REQuests	Number of file requests
SUMmary	TRANsaction	I/Otime	Average time spent waiting for I/O
SUMmary	TRANsaction	LIFETIME	Average transaction lifetime

Record Type	Class	Variable Name	Description
SUMmary	TRANsaction	PGMtime	Average time spent in program code, including both user and CICS code
SUMmary	TRANsaction	RSCStime	Average time spent waiting for any resource
SUMmary	TRANsaction	STGSUSPtime	Average time spent suspended due to storage
SUMmary	TRANsaction	SUSPtime	Average time spent suspended
SUMmary	TRANsaction	TCPUtime	Total CPU seconds for the transaction, including user, CICS, and Unicenter CA-Explore time
SUMmary	TRANsaction	THRESHOLDS	Number of thresholds exceeded
SUMmary	TRANsaction	TMIOtime	Average time spent waiting for terminal
SUMmary	TRANsaction	TRACetime	Average time spent in trace code
SUMmary	TRANsaction	USE	Number of transactions executed
SUMmary	TRANsaction	WAITtime	Average time waiting for an event control block
SUMmary	TRANsaction	WTRtime	Average time spent waiting to run

ID Options

The following ID options can be used with any SUMmary TRANsaction variable:

- APPLID
- CICSID
- ENDTIME
- JOBNAME
- OPERID
- STARTTIME
- TERMID
- TRANID
- USERID
- VMID
- VTAMLU

SYStem Record Type

Variable class descriptions and the available ID options for the System Record Type are discussed in this section.

THRESHOLD Variable Class

Record Type	Class	Variable Name	Description
SYStem	THRESHOLD	CLASS	Threshold class (COUNT, RATE, TIME, or PCT)
SYStem	THRESHOLD	COUNT	Number of thresholds exceeded
SYStem	THRESHOLD	DATE	The date that the threshold was triggered, in <i>mm/dd/yy</i> format
SYStem	THRESHOLD	LIMIT	Threshold limit value
SYStem	THRESHOLD	MAXVALUE	Maximum value that exceeded the limit
SYStem	THRESHOLD	MINVALUE	Minimum value that exceeded the limit
SYStem	THRESHOLD	NAME	The name of the threshold
SYStem	THRESHOLD	RESOURCE	The resource associated with the threshold
SYStem	THRESHOLD	TERMinal	The terminal ID (if the threshold is task related)
SYStem	THRESHOLD	TIME	The time that the threshold was triggered, in <i>hh:mm:ss</i> format
SYStem	THRESHOLD	TRANsaction	The transaction ID (if the threshold is task related)
SYStem	THRESHOLD	TYPE	The threshold type (UPPER or LOWER)
SYStem	THRESHOLD	USER	The user ID (if the threshold is task related)
SYStem	THRESHOLD	VALUE	Average value that exceeded the limit

ID Options

The following ID options can be used with any SYStem THRESHOLD variable:

- CICSID
- ENDTIME
- JOBNAME
- STARTTIME
- THRCLAS
- THRSID
- THRTYPE
- TRSCEID

Variable Options

Two kinds of options can be used with variables:

- Variable options
- ID options

These options can be used with most variables:

- COUNT
- FORMAT
- LOGICAL
- MAX and MIN
- MAXDATE and MAXTIME
- PerCenT
- RATE
- SELZERO
- TOTAL
- XTIME, XTIMEM, and NXTIME
- XTND and NXTND

The following options are valid only in tabular reports; they are discussed in the chapter titled “Tabular Reports.”

- DIFF
- RANGE
- RIGHT
- SELECT

ID options are explained in the chapter titled “ID Options.”

Specifying Options

Specify variable options following the variable to which they are to apply, as follows:

variable [*option* [*option* ...]]

Descriptions of Variable Options

This section provides a description of each variable option, including the syntax, operands, and examples.

COUNT Option

Use the COUNT option to convert a variable into the number of different identifiers that correspond to that variable.

Option Syntax and Operand

COUNT=*idoption*

Operand	Meaning
<i>idoption</i>	Any ID option valid for the variable

The following command would produce a tabular report displaying the number of terminals that had any activity:

```
TAB TRAN USE COUNT=TERMID
END
RUN
```

FORMAT Option

Use the FORMAT option to define the length of an output field or to specify how many decimal positions it contains.

The FORMAT option can also be used with ID options to control the number of characters printed for each identifier. For more information about formatting identifiers, refer to the chapter titled "ID Options."

Option Syntax and Operand

FORMAT (*outputformat*)

Operand	Meaning
<i>outputformat</i>	<p>One of the following, for numeric or alphanumeric output:</p> <ul style="list-style-type: none"> For numeric data, specify up to 15 Ns with or without a period to indicate the placement of the decimal point. For example, FORMAT(NN.NNN) indicates a five-digit output rounded to three decimal places. You can include any constants other than the letters N and C in the string. For example, specify FORMAT(NN.NNN%) to include a percent sign with each value in your report. For alphanumeric data, specify <i>n</i>C, where <i>n</i> is the number of characters to appear in the output. For example, FORMAT(4C) indicates an output of four characters. There is no limit on the value of <i>n</i>.

Note: If a variable output value is greater than the output format you specify, the value is replaced by an asterisk (*) in your report. For example, if you specify FORMAT(NN.NN) with a variable, all values greater than 99.99 for that variable are replaced by asterisks.

The following table shows examples of the FORMAT option and the result of each:

Sample Entry	Result
TRAN LIFE FORMAT(NNN.NN)	By default, the output of the variable TRAN LIFE appears in the format N.NNN, NN.NN, or NNNNN, depending on the size of the value. The FORMAT option here forces the output to two decimal places.
TRAN USE RATE FORMAT(NN.N)	The RATE option used with the variable TRAN USE defaults to two decimal positions. The FORMAT option here alters the output to one decimal place.

LOGICAL Option

By default, the report writer checks all input records and discards any with negative values. Use the LOGICAL option with a variable to bypass this check and treat all values for the variable as unsigned (logical) values.

Option Syntax

LOGICAL

The following commands create a tabular report on terminals, treating all values for the variable TERM USE as unsigned. Records with a negative value for TERM RESP will still be discarded.

```
TAB TERMID TERM RESP AND,  
      TERM USE LOGICAL  
END  
RUN
```

MAX and MIN Options

Use the MAX and MIN options to force a variable that by default prints a count or an average to print the minimum or maximum value.

Option Syntax and Operand

MAX[=*idoption*]

MIN[=*idoption*]

Operand	Meaning
<i>idoption</i>	Any ID option valid for the variable. The values produced by the MAX and MIN options are largely dependent on the structure of the data records. Use this operand to control how the values are computed.

The following examples use the MAX and MIN options:

```
TRAN LIFE MAX
```

```
TRAN LIFE MIN
```

```
DATASET USE MAX=TRANNUM
```

TRAN LIFE is defined as the average transaction lifetime. TRAN LIFE MAX prints the maximum transaction lifetime. TRAN LIFE MIN prints the minimum transaction lifetime.

DATASET USE is defined as the number of file requests. DATASET USE MAX=TRANNUM prints the maximum number of file requests for any single transaction number.

MAXDATE and MAXTIME Options

Use the MAXDATE and MAXTIME options to print the date and time when the value of a variable is highest.

Option Syntax

```
MAXDATE  
MAXTIME
```

The following examples use the MAXDATE and MAXTIME options with the variable TRAN LIFE:

```
TRAN LIFE MAXDATE  
TRAN LIFE MAXTIME
```

TRAN LIFE is defined as the average transaction lifetime. TRAN LIFE MAXDATE prints the date on which transaction lifetime was highest. TRAN LIFE MAXTIME prints the time at which this value was highest.

PerCenT Option

Use the PerCenT option to cause a variable to represent a resource's usage as a percentage of the total value for the system. You can use the PerCenT option only with variables that represent a total that is **not** the same as the value for the entire system.

Option Syntax

```
PerCenT
```

In the following example, the PLOT command plots the use of the transaction Cinq against the total system transaction use. The second command, PLOT2, plots the *percentage* of transaction use for Cinq compared to the total system transaction use.

```
PLOT TRANSACTION USE TRANID(CINQ)  
PLOT2 TRANSACTION USE PCT TRANID(CINQ)  
END  
RUN
```

RATE Option

Use the RATE option to display a rate (number per second, minute, hour, day, or week) rather than a count. You can use the RATE option only with variables that represent a count.

Option Syntax and Operands

RATE[(*period* [, WEIGHTED] [*type*])]

Operand	Meaning
<i>period</i>	<p>Defines the time interval the rate is to be based upon. Specify one of the following:</p> <p>RPTINT Calculates the rate based on the reporting interval. This is the default.</p> <p>SYSINT Calculates the rate based on the interval of the Unicenter CA-Explore main system record, thereby accounting only for the time Unicenter CA-Explore was active logging data.</p> <p>[<i>nn</i>] interval Calculates the rate based on the interval specified. <i>nn</i> can be 1 to 99; the default is 1. Replace <i>interval</i> with SECONDS, MINUTES, HOURS, DAYS, or WEEKS.</p>
WEIGHTED	Prints rates based only on those intervals that show activity for the specified variable.
<i>type</i>	<p>Specify one of the following:</p> <p>COUNT Prints the number of active intervals.</p> <p>MAX Prints the maximum value for any one interval.</p> <p>MAXDATE Prints the date of the interval during which the maximum value occurred.</p> <p>MAXRATE Prints the maximum rate for any one interval.</p> <p>MAXTIME Prints the time of the interval during which the maximum value occurred.</p>

In the following example, the PLOT command plots the use of the transaction CINQ against the total system transaction use, and the PLOT2 command plots the transaction rate for CINQ compared to the total system transaction rate:

```
PLOT TRAN USE      TRANID(CINQ)
PLOT2 TRAN USE RATE TRANID(CINQ)
  EACH HOUR
END
RUN
```

The following commands create a tabular report showing the hourly transaction rate (average number per hour) each day:

```
TAB DATE TRAN USE RATE(1 HOUR)
  EACH DAY
END
RUN
```

The following commands create a tabular report showing the highest hourly transaction rate for each day:

```
TAB DATE TRAN USE RATE(1 HOUR,MAXRATE)
  EACH DAY
END
RUN
```

SELZERO Option

By default, the report writer checks all input records and discards any with zero activity for the specified variables. For example, a tabular report on terminal activity includes only those terminals that have a positive value for at least one variable specified with the TAB command. Use the SELZERO option to bypass this check and include all resources in the report.

Option Syntax

```
SELZERO
```

By default, a graph of file requests by transaction ID includes only transactions that issued at least one file request. The following commands graph the average number of file requests by every transaction, including those with no file requests:

```
GRAF TRANID DATASET USE SELZERO
END
RUN
```

TOTAL Option

Use the TOTAL option to force a variable that by default prints an average to print a total.

Option Syntax

```
TOTAL
```

The following example uses the TOTAL option with the variable TRAN CPU:

```
TRAN CPU TOTAL
```

By default, the variable TRAN CPU creates a column showing the average CPU time per transaction. Specifying the variable with the option TOTAL creates a column showing the total CPU time used by all transactions.

XTIME, XTIMEM, and NXTIME Options

By default, all time-based report writer variables are displayed in seconds. If you expect a variable to show a time of many hours, you can use the XTIME option to translate the time to *hhh:mm:ss* format, where *hhh* is hours, *mm* is minutes, and *ss* is seconds.

If you want more precise output of time values, you can use the XTIMEM option or the NXTIME option. The XTIMEM option translates the time value to *hhh:mm:ss:lll* format, where *hhh* is hours, *mm* is minutes, *ss* is seconds, and *lll* is milliseconds. The NXTIME option changes the value to *ssss:lll* format, where *ssss* is seconds and *lll* is milliseconds.

The *hhh:mm:ss*, *hhh:mm:ss:lll*, and *ssss:lll* formats take up more column space than the default format.

Option Syntax

```
XTIME
XTIMEM
NXTIME
```

The following example uses the XTIME option with the variable TRAN TCPU:

```
TRAN TCPU XTIME
```

XTND and NXTND Options

Use the XTND (extend) and NXTND options to increase or decrease the number of digits in a variable's output.

Most variables have a maximum value of five digits. If a number is larger than 99999, the report writer divides it by 1000 and adds the letter K (for example, 200,000 becomes 200K). If the number is still too large, the report writer divides by 1000 again and adds the letter M, and so on. For values representing storage, the report writer divides large values by 1024 and adds the letter k; if the number is still too large, it divides by 1024 again and adds the letter m.

The XTND and NXTND variable options are **not** valid for variables that yield time values. You can use the XTIME, XTIMEM, and NXTIME options to control the output of time variables, as described previously in this chapter.

Use the XTND option to support the printing of up to eight digits. This allows you to print more precise output of numbers up to 99,999,999 (including commas).

XTND Option Syntax

XTND

You might use the XTND option with the variable TRAN USE if you expect the number of transactions to be more than 99999 and you want an exact count, as follows:

```
TRAN USE XTND
```

Some numbers that the report writer expects to be large can be extended by default. Use the NXTND option to reduce the number of digits printed, thus increasing the number of columns that can fit across a page.

NXTND Option Syntax

NXTND

The following example uses the NXTND option with the variable TRAN USE:

```
TRAN USE NXTND
```

This chapter provides descriptions of ID options, their uses, and their identifiers.

You can use ID options with certain commands, with variables, or with both, as follows:

- Use ID options with commands to organize the data in a tabular, graph, or flashback report by a specific type of resource. For example, you can create a tabular report organized by transaction ID, in which each row contains information about only a single transaction.
- You can also specify one or more identifiers to limit the report to the particular resource or resources specified by the identifiers.
- Use ID options with variables in any type of report to limit the range of the variable to a particular resource or group of resources. For example, you can limit the range of a variable to a specific transaction or a specific group of programs.
- When you use ID options with variables, you generally must include one or more identifiers that specify the particular resources on which you want information.

Descriptions of ID Options

The following table lists all ID options, the identifiers that are valid for each, their maximum length in characters, and the variable record types with which they can be used:

ID Option	Identifiers You Can Specify with the ID Option	Length of Identifiers	Variable Classes You Can Use with the ID Option
ABCODE	Abend codes for transactions	4	Any PERFormance variable
APPLID	VTAM APPLIDs of CICS regions of CICS systems	8	Any PERFormance variable Any SUMmary variable
CICSID	CICS region IDs for CICS systems. Region IDs are set with EXPCMIT, the Unicenter CA-Explore for CICS online monitor initialization table.	2	Any CICS variable
DCTID	Dataset names of transient datasets	4	PERFormance TERMinal PERFormance TRANsaction PERFormance TRANSIENT
DLIPOOL	DL/I pool keys for DL/I files	1	Some PERFormance DLI, as described later in this chapter. PERFormance TERMinal PERFormance TRANsaction
DLITYPE1	DL/I type1 for DL/I files. Valid values are: D DL/I R Remote ? Unknown	1	PERFormance DLI PERFormance TERMinal PERFormance TRANsaction
DLITYPE2	DL/I type2 for DL/I files. Valid values are: A DL/I ACB D DL/I DBD F DL/I function M Module or program P DL/I PSB ? Unknown	1	Some PERFormance DLI, as described later in this chapter PERFormance TERMinal PERFormance TRANsaction

ID Option	Identifiers You Can Specify with the ID Option	Length of Identifiers	Variable Classes You Can Use with the ID Option
EIPID	Command-level request types for command level interface data	8	PERformance COMMAND PERformance TERMinal PERformance TRANsaction
ENDTIME	None	8 (<i>hh:mm:ss</i> format)	All variable classes
EXITID	User exit names	8	PERformance EXIT PERformance TERMinal PERformance TRANsaction
FILEID	File names	8	INTerval DATASET PERformance DATASET PERformance DLI PERformance SQL PERformance TERMinal PERformance TRANsaction
FTYPE1	File type1. Valid values are: B BDAM D Data table E DATACOM or SUPRA R Remote T TOTAL database U User file V VSAM ? Unknown	1	PERformance DATASET PERformance TERMinal PERformance TRANsaction
FTYPE2	File type2. Valid values are: B BDAM keyed E ESDS K KSDS L LDS N BDAM non-keyed R RSDS S System data table U User data table ? Unknown	1	PERformance DATASET PERformance TERMinal PERformance TRANsaction
JOBNAME	CICS jobnames	8	Any CICS variable

ID Option	Identifiers You Can Specify with the ID Option	Length of Identifiers	Variable Classes You Can Use with the ID Option
JOURNID	Journal IDs	8	PERformance JOURNAL PERformance TERMinal PERformance TRANsaction
LSRID	LSR pool IDs for LSR data	1	INTerval DATASET
NETNAME	VTAM logical unit names for terminals	8	Any PERformance variable
OPERID	Operator IDs	3	Any PERformance variable Any SUMmary variable
OPERNAME	Operator names	20	Any PERformance variable (version 4 only)
PROGID	Program names	8	PERformance DATASET PERformance PROGram PERformance TERMinal PERformance TRANsaction
RSCSID	Exit resource names	8	PERformance RESOURCE PERformance SECURITY PERformance TERMinal PERformance TRANsaction
SQLID	SQL/DS classifications, program names, and package names	8	PERformance SQL PERformance TERMinal PERformance TRANsaction
SQLTYPE	SQL/DS resource types. Valid values are: A Access module P Program	1	PERformance SQL PERformance TERMinal PERformance TRANsaction
STARTTIME	None	8 (<i>hh:mm:ss</i> format)	All variable classes
STORID	Storage types for transaction storage	8	PERformance STORAGE PERformance TERMinal PERformance TRANsaction

ID Option	Identifiers You Can Specify with the ID Option	Length of Identifiers	Variable Classes You Can Use with the ID Option
SYSIDNT	System IDs of CICS systems	8	PERformance IRC PERformance ISC PERformance TERMinal PERformance TRANsaction
TERMID	Terminal IDs	4	Any PERformance variable Any SUMmary variable
THRCLAS	Threshold classes. Valid values are: COUNT RATE TIME PCT	5	PERformance TERMinal PERformance THRESHOLD PERformance TRANsaction SYStem THRESHOLD
THRSID	Threshold names	8	PERformance TERMinal PERformance THRESHOLD PERformance TRANsaction SYStem THRESHOLD
THRSSID	Secondary threshold names	8	PERformance TERMinal PERformance THRESHOLD PERformance TRANsaction SYStem THRESHOLD
THRTYPE	Threshold types. Valid values are: UPPER LOWER	5	PERformance TERMinal PERformance THRESHOLD PERformance TRANsaction SYStem THRESHOLD
TRANCLASS	Transaction class. Valid values are: L Long M Medium N Non-terminal S Short	1	Any PERformance variable
TRANID	Transaction IDs	4	Any PERformance variable Any SUMmary variable

ID Option	Identifiers You Can Specify with the ID Option	Length of Identifiers	Variable Classes You Can Use with the ID Option
TRANNUM	Transaction numbers	7 exactly	Any PERFormance variable
TRSCEID	Threshold resource names	8	PERFormance TERMinal PERFormance THRESHOLD PERFormance TRANsaction SYSem THRESHOLD
TSTGID	Temporary storage IDs for temporary storage data	8	PERFormance TEMPSTOR PERFormance TERMinal PERFormance TRANsaction
USERID	User IDs	8	Any PERFormance variable Any SUMmary variable
UTRANID	Umbrella names	32	Any PERFormance variable
VMID	VM machine IDs of VSE guest machines	8	Any INTerval variable Any PERFormance variable Any SUMmary variable
VOLSER	Volume serial number	6	PERFormance DATASET
VTAMLU	VTAM logical unit names of terminals	8	Any PERFormance variable Any SUMmary variable
WAITID	Wait types	8	PERFormance TERMinal PERFormance TRANsaction PERFormance WAIT

Using ID Options With Commands

Use ID options with commands to organize the data in a report by a specific type of resource, such as transaction, terminal, program, or user.

You can use ID options with the TAB, TAB2, GRAF, and FLASHBACK commands.

You cannot use ID options with any of the plot commands. You can, however, use ID options with variables in plot reports, as described later in this chapter.

To use an ID option with a command, specify the ID option after the command and before any variables.

Specifying ID Options

The syntax of ID options when used with commands is as follows:

```
idoption[(identifier[, identifier...])]
```

The operands for ID options have the following meanings:

Operand	Meaning
<i>idoption</i>	The ID option you specify determines the type of resource the resulting report will be organized by.
<i>identifier</i>	An individual resource or group of resources the report is to be limited to. The types of identifiers that are valid for each ID option are given earlier in this chapter.

You are not required to specify identifiers when you use an ID option with a command. If you specify one or more identifiers with an ID option, the resulting report will not only be organized by the type of resource specified by the ID option, but the entire report will be limited to the resources specified by the identifiers.

Examples

The following TAB command includes the ID option TRANID to generate a report organized by transaction ID that shows the average lifetime and total usage of transactions:

```
TAB TRANID TRAN USE AND TRAN LIFE  
END  
RUN
```

The preceding commands generate a report like the following:

TRAN ID	TRAN USE	TRAN LIFE
FXFR	2482	0.361
PSPI	939	0.248
CSNE	699	0.026
POSS	610	0.120
X25T	546	0.818
RAPL	474	0.363
WISB	313	18.19
WISI	263	5.077
TCP	224	912.2
KCP	224	912.2
RAPP	215	40.15
CHAI	160	0.193
CSPG	157	0.032
CSPQ	137	0.060
CATD	125	0.465
VGBC	115	17.02
CSGM	107	1.030
CSSN	84	35.77
CSPS	71	0.057
PADI	67	16.81
SUMMARY	8012	53.98

TAB Report Using an ID Option and Identifiers

The following TAB command is similar to the preceding example, but the ID option TRANID is specified with the identifier C* to generate a report on a subset of the transactions in the preceding report:

```
TAB TRANID(C*) TRAN USE AND TRAN LIFE  
END  
RUN
```

The preceding commands generate a report like the following:

TRAN ID	TRAN USE	TRAN LIFE
CSNE	699	0.026
CHAI	160	0.193
CSPG	157	0.032
CSPQ	137	0.060
CATD	125	0.465
CSGM	107	1.030
CSSN	84	35.77
CSPS	71	0.057
SUMMARY	1540	2.104

TAB and TAB2 Report Using ID Options

The following example uses the ID option TERMID with the TAB command and the ID option TRANID with the TAB2 command to create a report on terminals and the transactions associated with them:

```
TAB TERMID TRAN USE AND TRAN LIFE
TAB2 TRANID TRAN USE AND TRAN LIFE
END
RUN
```

The preceding commands generate a report like the following:

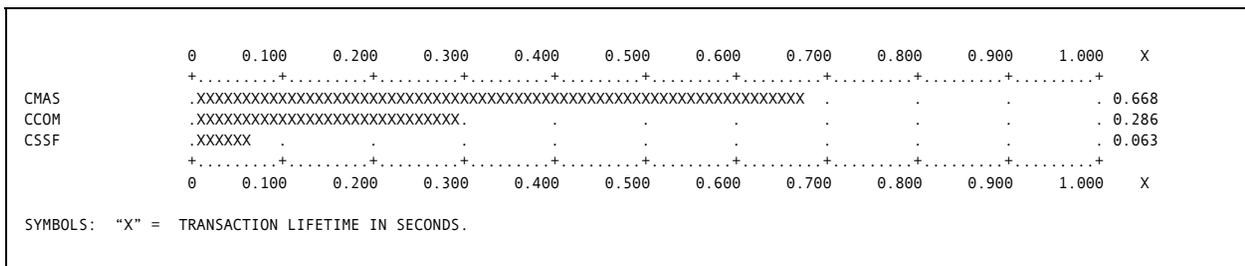
TERM ID	TRAN USE	TRAN LIFE	TRAN ID	TRAN USE	TRAN LIFE
F014	2483	0.361	FXFR	2482	0.361
			CSGM	1	0.012
			SUMMARY	2483	0.361
N/A	2160	326.2	CSNE	699	0.026
			RAPL	474	0.363
			KCP	224	912.2
			TCP	224	912.2
			CSPQ	137	0.060
			CATD	125	0.465
			CSPS	71	0.057
			JJJ	61	3161
			DESP	52	2.587
			UPIK	44	3.009
			SUMMARY	2111	4704K
0A12	401	1.754	WISI	222	0.579
			CHAI	103	0.217
			CSPG	51	0.032
			PADI	15	36.67
			CSAC	4	0.047
			CSGM	3	0.052
			ENDD	2	0.172
			END	1	0.024
			SUMMARY	401	1.754
SUMMARY	5044	140.0		8684	87.43

Graph Report Using an ID Option and Identifiers

The following GRAF command is specified with the ID option TRANID and the identifiers CMAS, CCOM, and CSSF, which limits the resulting report to data about only those transactions:

```
GRAF TRANID(CMAS,CCOM,CSSF) TRAN LIFETIME SCALE(1.0)
  FROM 06/06/99 00.00.00
  TO   06/07/99 00.00.00
END
RUN
```

The preceding commands generate a report like the following:



Flashback Report Using an ID Option and Identifiers

The following commands produce a FLASHBACK LIST report showing the transactions of operator SWS only:

```
FLASHBACK LIST OPERID(SWS)
END
RUN
```

The preceding commands generate a report like the following:

CICS ID	END TIME	TERM ID	OPER ID	TRAN ID	TRAN NUMBER	TRAN LIFE	TRAN CPU	TERM RESP	TERM IO	TRAN I/O	FILE COUNT	FILE USE	PROG COUNT	ABEND CODE
B1	00.45.29	0A02	SWS	POSS	67	1.011	0.274	1.011	2	0.000	0	0	3	
B1	00.45.35	0A02	SWS	POSS	69	2.387	0.051	1.187	3	0.199	1	2	6	
B1	00.45.38	0A02	SWS	POSS	70	0.419	0.249	0.000	1	0.000	0	0	3	
B1	00.45.44	0A02	SWS	DEDA	71	5.040	0.705	0.992	4	0.034	1	5	4	
B1	00.45.44	0A02	SWS	POSS	72	0.078	0.023	0.034	2	0.039	1	1	4	
B1	00.46.03	0A02	SWS	POSS	73	0.025	0.002	0.025	2	0.000	0	0	3	
B1	00.46.12	0A02	SWS	SPAM	75	1.422	0.291	0.462	3	0.040	1	1	4	
B1	00.46.18	0A02	SWS	SPAM	76	0.446	0.085	0.199	3	0.052	1	1	4	
B1	00.46.37	0A02	SWS	SPAM	77	0.094	0.020	0.000	1	0.048	1	1	4	
B1	00.46.38	0A02	SWS	POSS	78	0.088	0.025	0.033	2	0.038	1	1	4	
B1	00.46.40	0A02	SWS	POSS	79	0.034	0.021	0.000	1	0.000	0	0	3	
B1	00.58.19	0A02	SWS	PBLM	80	699.2	1.925	0.109	156	0.030	1	1	4	
B1	00.58.19	0A02	SWS	POSS	81	0.091	0.026	0.041	2	0.047	1	1	4	
B1	05.28.09	0A02	SWS	POSS	574	0.028	0.017	0.000	1	0.000	0	0	3	
B1	05.28.09	0A02	SWS	PSPI	575	0.047	0.016	0.009	2	0.000	0	0	3	
B1	05.28.19	0A02	SWS	PSPI	577	0.177	0.054	0.044	4	0.000	0	0	3	
B1	05.28.29	0A02	SWS	PSPI	580	0.170	0.049	0.029	5	0.000	0	0	3	
B1	05.28.41	0A02	SWS	PSPI	581	0.152	0.050	0.024	5	0.000	0	0	3	
B1	05.28.53	0A02	SWS	PSPI	582	0.148	0.049	0.024	5	0.000	0	0	3	
B1	05.29.03	0A02	SWS	PSPI	584	0.154	0.049	0.025	5	0.000	0	0	3	
B1	05.29.20	0A02	SWS	PSPI	588	0.118	0.040	0.018	5	0.000	0	0	3	
B1	05.29.37	0A02	SWS	PSPI	592	0.100	0.045	0.013	5	0.000	0	0	3	
B1	05.29.48	0A02	SWS	PSPI	593	0.130	0.046	0.018	5	0.000	0	0	3	
B1	05.32.32	0A02	SWS	PSPI	603	0.219	0.051	0.039	5	0.000	0	0	3	
B1	05.34.14	0A02	SWS	PSPI	606	0.129	0.032	0.129	2	0.000	0	0	3	
-----					05.34.14	8003	28.47	0.168	0.132	231	0.021	0	10	3

Using ID Options with Variables

Use ID options with individual variables to limit the range of the variable to a particular resource or group of resources. You can use ID options with variables in any type of report.

When you use ID options with variables, you must specify one or more identifiers.

To use an ID option with a variable, specify the ID option and its identifiers immediately following the variable you want to restrict.

Specifying ID Options with Variables

The syntax of ID options when used with variables is as follows:

```
idoption (identifier[, identifier...])
```

The operands for ID options have the following meanings:

Operand	Meaning
<i>idoption</i>	<p>The ID option you specify determines the type of resource the report is to be about.</p> <p>The ID options that can be used with a given variable depend upon the variable's class and record type. The ID options that are valid for each class and record type are given in the chapter titled "Variables."</p>
<i>identifier</i>	<p>An individual resource or group of resources the variable is to be limited to. The types of identifiers that are valid for each ID option are given earlier in this chapter.</p>

Using ID Options With Commands Versus With Variables

The following examples illustrate the difference between using ID options with commands and using ID options with variables.

The following commands generate a report showing the number and percentage of transactions used by each of two CICS systems. The ID option and identifiers CICSID(A1,B1) are specified with the TAB command.

```
TAB DATE CICSID(A1,B1) TRAN USE AND TRAN USE PCT
      EACH DAY
      FROM 12/08/98
      TO   12/09/98
END
RUN
```

The following commands also generate a report showing the number and percentage of transactions used by each of two CICS systems. However, in this example, the ID option and identifier CICSID(A1) and the ID option and identifier CICSID(B1) are specified with variables.

```
TAB DATE TRAN USE          HDR1(-TOTAL-) HDR2(TRAN USE) AND,
      TRAN USE CICSID(A1)  HDR1(-A1-)   HDR2(TRAN USE) AND,
      TRAN USE CICSID(A1) PCT HDR1(-A1-)   HDR2(% USE)   AND,
      TRAN USE CICSID(B1)  HDR1(-B1-)   HDR2(TRAN USE) AND,
      TRAN USE CICSID(B1) PCT HDR1(-B1-)   HDR2(% USE)

      EACH DAY
      FROM 12/08/98
      TO   12/09/98
END
RUN
```

Sample Report 1

The first of the two preceding sets of sample commands generates the following report:

DATE	CICS ID	TRAN USE	TRAN % USE
12/08/98	B1	368	71.0
	A1	150	29.0
	SUM	518	100.0
12/09/98	B1	5488	79.8
	A1	1393	20.2
	SUM	6881	100.0

Sample Report 2

The second of the two preceding sets of sample commands generates the following report:

DATE	-TOTAL- TRAN USE	-A1- TRAN USE	-A1- % USE	-B1- TRAN USE	-B1- % USE
12/08/98	518	150	29.0	368	71.0
12/09/98	6881	1393	20.2	5488	79.8
PERIOD	7399	1543	20.9	5856	79.1

Note how the preceding reports differ:

- The first report is organized by both time and CICS ID, and the entire report is restricted to data about the CICS systems A1 and B1.
- The second report is organized only by time, and each ID option restricts the range of just the one variable it is specified with. For example, the third column in the report, which was generated by the variable and ID option TRAN USE CICSID(A1), contains information about system A1 only.

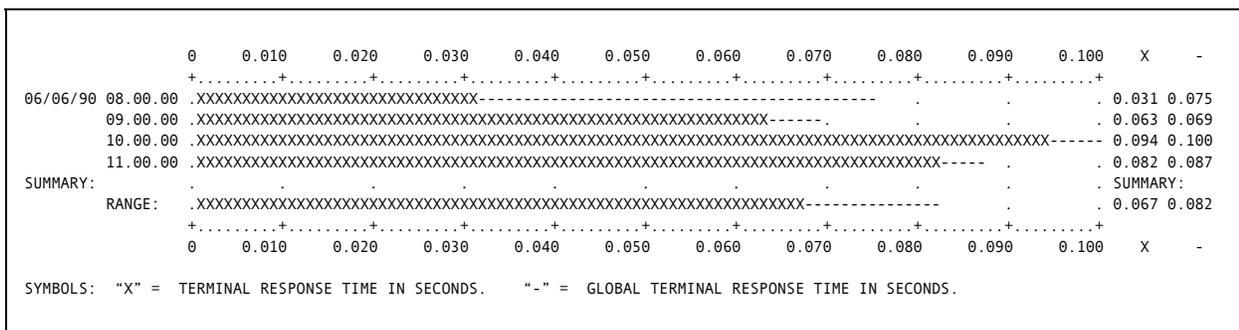
Example Using ID Options with Variables

Plot Report

The following commands generate a plot report that shows the response time for terminal L310:

```
PLOT TERM RESP TERMID(L310)
END
RUN
```

The preceding commands generate a report like the following. In addition to the response time for terminal L310, the response time for all terminals is plotted, allowing you to compare the performance of terminal L310 to the performance of the rest of the terminals in the system.



Using ID Options With Both Commands and Variables

A single report writer command can include more than one ID option. You can specify an ID option with the command, and specify ID options with one or more of the variables included in the command.

The following TAB command is specified with the ID option TRANID; this ID option organizes the report by transaction ID. In addition, one of the variables is specified with the ID option and identifier TRANID(CINQ); this ID option restricts the range of that variable to the transaction CINQ.

```
TAB TRANID TRAN USE AND,  
      TRAN LIFETIME AND,  
      TRAN MAXLIFETIME AND,  
      TRAN USE TRANID(CINQ) HDR1( TRAN USE ) HDR2(FOR TRAN "CINQ")  
FROM 06/06/99 08:00:00  
TO 06/06/99 12:00:00  
END  
RUN
```

The preceding commands generate a report similar to the following:

TRAN ID	TRAN USE	TRAN LIFETIME	MAX TRAN LIFETIME	TRAN USE FOR TRAN "CINQ"
CINQ	1081	0.764	102.5	1081
CMAS	651	1.580	115.2	0
PINQ	651	0.564	53.06	0
IINQ	365	0.443	67.73	0
SUMMARY	2748	0.867	102.5	1081

Note that the last column of the report has only one nonzero entry. Since this report is organized by the ID option TRANID, each row in the report shows information about only the transaction listed in the first column. Because the last column of the preceding report represents the use of transaction CINQ only, only the detail line for transaction CINQ has a nonzero value in the last column.

Specifying Identifiers with ID Options

Identifiers specify particular resources of the type determined by the ID option. The effect of specifying an identifier depends on whether the ID option it is used with is specified with a command or with a variable, as follows:

- If an ID option is used with a command, specifying identifiers restricts the report to information about the particular resources specified.
- If an ID option is used with a variable, specifying identifiers restricts the range of the variable to the particular resources specified.

The identifiers that are valid with each ID option are listed earlier in this chapter.

Identifiers can specify one of the following:

- An individual resource of the type determined by the ID option. For example, the ID option OPERID can be followed by an identifier that names a specific operator, that is, an operator ID.
- A group of resources of the type determined by the ID option. You can specify more than one resource in two ways:
 - You can list several individual resources separated by commas.
 - You can specify one or more groups of resources using *generic characters*, as explained in the next section. For example, the ID option TERMID can be followed by an identifier that specifies the group of terminals having IDs beginning with the letter C.

Using Generic Characters in Identifiers

Identifiers can contain the following generic characters to report on groups of resources:

- A plus sign (+) tells the report writer not to compare that position in the identifier. A place in an identifier filled in with the plus sign (+) is always a match.
- An asterisk (*) tells the report writer that all places after and including the asterisk are matches.
- A not sign (-) tells the report writer to exclude the identifier that follows it. You can use the generic characters * and + when specifying the identifier to be excluded.

The following GRAF command produces a report on terminal use for only certain transactions:

```
GRAF TRANID(C*,-C100, S104) TERMINAL USE
```

In this example, the ID option TRANID is specified with three identifiers, C*, -C100, and S104. The identifier C* includes all transactions starting with C. The identifier -C100 excludes the transaction C100. The identifier S104 includes the transaction S104. Therefore, this GRAF command generates a report showing terminal use for transactions starting with C except transaction C100, and for the transaction S104.

Specifying Hexadecimal Identifiers

Identifiers can be specified in their hexadecimal representation by placing the hexadecimal value inside single quotes.

The following two commands generate the same report, a report on only those transactions that have IDs beginning with EXP:

```
TAB TRANID(EXP*) CICS TRAN USE  
END
```

```
TAB TRANID('C5E7D7'*) CICS TRAN USE  
END
```

Formatting Identifiers in Output

In some situations, you may want to change the way identifiers are printed. You can do this in two ways:

- You can use the `FORMAT` option to limit the number of characters printed for each identifier.
- You can use the `DEFINE` command to define new identifiers or redefine existing identifiers.

Each of these methods is explained in more detail in the following sections.

Using the `FORMAT` Option

You can use the `FORMAT` option to limit the number of characters printed for each identifier. Truncation of the identifier names is done just before printing. Therefore, even if two identifiers are identical after truncation, their data will not be combined in the report.

See the chapter titled “Variable Options” for a complete description of the `FORMAT` option and its full syntax.

Option Syntax and Operand

The syntax of the `FORMAT` option when used with ID options is as follows:

```
idoption FORMAT(nC)
```

The operand for the `FORMAT` option is as follows:

Operand	Meaning
<i>n</i>	The number of characters to appear in the output. For example, <code>FORMAT(4C)</code> indicates an output of four characters. There is no limit on the value of <i>n</i> .

Assume, for example, that your user IDs are eight characters long. A portion of a report on users might look like this:

USER ID	TRAN USE	TRAN ABENDS	TRAN CPU	TRAN TCPU	TRAN LIFE	TRAN MAX LIFE	TRAN STOR	TRAN WAITS	FILE USE	FILE WAITS	PROG USE	PROG ABENDS	TERM WRITES
AALBERTS	4	0	0.024	0.097	0.233	0.749	8934	3	2	0	16	0	5
AANDERSO	3	0	0.016	0.049	0.110	0.262	11872	3	2	0	15	0	4
ABROWN	9	2	0.070	0.630	83.11	370.5	8150	4	1	0	51	2	14
ABROWNIN	63	0	0.037	2.321	1.917	19.00	7665	5	158	0	198	0	44
ACHRISTI	3	1	0.082	0.247	123.7	370.8	11125	5	1	0	18	1	5
AKENNEDY	19	0	0.166	3.150	20.43	347.5	34520	41	710	0	61	0	16
ALEWIS	4	0	0.859	3.437	6.840	13.18	116K	13	10	0	176	0	4
APETERSO	12	0	0.080	0.961	0.240	0.472	22526	6	2	0	66	0	32
BAUSTEN	48	1	0.109	5.245	16.79	715.1	37487	10	11	0	304	1	56

The user IDs in the preceding report could be shortened using the FORMAT option, saving enough column space to include another variable. The following commands are similar to those used to generate the preceding report, but they include the option FORMAT(4C) and the additional variable TERM RESP:

```
TAB USERID FORMAT(4C) FOR 10 SORT ALPHA,
  TRAN USE AND TRAN ABENDS AND,
  TRAN CPU AND TRAN TCPU AND,
  TRAN LIFE AND TRAN MAXLIFE AND,
  TRAN STOR AND TRAN WAITS AND,
  DATASET USE AND DATASET WAITS AND,
  PROGRAM USE AND PROGRAM ABENDS AND,
  TERM USE AND TERM RESP
END
RUN
```

The preceding commands produce a report like the following:

USER ID	TRAN USE	TRAN ABENDS	TRAN CPU	TRAN TCPU	TRAN LIFE	TRAN MAX LIFE	TRAN STOR	TRAN WAITS	FILE USE	FILE WAITS	PROG USE	PROG ABENDS	TERM WRITES	TERM RESP
AALB	4	0	0.024	0.097	0.233	0.749	8934	3	2	0	16	0	5	0.179
AAND	3	0	0.016	0.049	0.110	0.262	11872	3	2	0	15	0	4	0.075
ABRO	9	2	0.070	0.630	83.11	370.5	8150	4	1	0	51	2	14	0.213
ABRO	63	0	0.037	2.321	1.917	19.00	7665	5	158	0	198	0	44	0.344
ACHR	3	1	0.082	0.247	123.7	370.8	11125	5	1	0	18	1	5	0.218
AKEN	19	0	0.166	3.150	20.43	347.5	34520	41	710	0	61	0	16	1.467
ALEW	4	0	0.859	3.437	6.840	13.18	116K	13	10	0	176	0	4	6.840
APET	12	0	0.080	0.961	0.240	0.472	22526	6	2	0	66	0	32	0.072
BAUS	48	1	0.109	5.245	16.79	715.1	37487	10	11	0	304	1	56	1.499

Note that in the preceding report, the user ID ABRO appears twice in the first column. This is because the first four characters of two user IDs were the same.

Using the DEFINE Command

You can use the DEFINE command to define new identifiers or redefine existing identifiers.

The DEFINE command can also be used to define new variables. For a complete description of the DEFINE command and its full syntax, refer to the chapter titled "Commands."

The syntax of the DEFINE command when used to define ID options based on existing ID options is as follows:

```
DEFINE ID newid [oldid [(start,length)]] ['string']
```

Operand	Meaning
ID	Requests the definition of a new ID option.
<i>newid</i>	The name of the new ID option you want to define.
<i>oldid</i>	The name of an existing ID option you want to be a component of the new ID option.
<i>start,length</i>	If you do not want to include the entire existing ID option in the new ID option, specify the starting position (<i>start</i>) and length (<i>length</i>) of the part of it you want to include.
<i>string</i>	A character string that you want to be a component of the new ID option. You must enclose the string in single quotes. The string can be more than one character in length, and it can include or consist entirely of blanks.

When defining a new ID option, you can include multiple character strings or existing ID options, in any order. When printed, components are concatenated without intervening spaces.

Use the following syntax to define new ID options without using existing ID options. This allows you to report on any data field located in any record. The syntax is shown on two lines with a continuation comma at the end of the first line.

```
DEFINE XID name PRODUCT=EXPC RECORD=rectype [SUBTYPE=subtype] OFFSET=data-offset LENGTH=data-length,
[TRIPLET=triplet-offset] [TYPE=data-type] [HDR1=header1] [HDR2=header2]
```

Using the DEFINE Command Versus the FORMAT Option

You should use care when using the DEFINE command to create shortened forms of identifiers because, unlike the FORMAT option, the DEFINE command can merge data about different identifiers. For instance, the DEFINE command could be used to redefine the ID option USERID to be the first four characters of the user ID. If this command had been used in the example earlier in this chapter instead of the FORMAT option, the data for ABROWN and ABROWNING would have been merged, since both user IDs would have been redefined to the same value, ABRO.

However, the DEFINE command offers more flexibility than the FORMAT option. For example, while the FORMAT option allows you to truncate identifiers and retain the first characters, the DEFINE command allows you to select any portion of the identifiers. For instance, in the example earlier, the ID option USERID could have been redefined to be four characters long beginning with the second character rather than with the first.

The following examples show ways to enter the DEFINE command and the result of each:

Sample Entry	Result
DEFINE ID OPERID OPERID(2,5)	Redefines the length of the ID option USERID to be 5 characters, starting with position 2.
DEFINE ID TRANID TRANID(1,4)	Redefines the length of the ID option TRANID to be 4 characters, starting with position 1.
DEFINE ID PROGID 'Program ' PROGID	Redefines the ID option PROGID to include the word Program and a blank before the program name.
DEFINE ID MYKEY TRANID ' ' TERMID	Creates a new ID option called MYKEY with components TRANID and TERMID and places a blank between the two components.
DEFINE XID KEY19 PRODUCT=EXPC RECORD=19, OFFSET=X14 LENGTH=8 TYPE=HEX	Creates the new ID option KEY19, which is of record type 19, at offset X'14', of length 8, and of type hexadecimal. The continuation comma allows the command to be continued on a second line.

Tabular Reports

Tabular reports organize data into rows and vertical columns. This chapter describes how to generate tabular reports using the TAB and TAB2 commands, and provides examples.

When you create tabular reports, you use *labels* or *keys* at the beginning of each row describe what the data in the line represents a time or a particular resource, such as a transaction, terminal, or user. *Headings* at the top of each column describe a variable; the value of that variable for each time or resource appears in that column.

For more information on related topics, refer to the following chapters:

- See the chapter titled “Commands” to become familiar with report writer commands.
- For an explanation of the structure of tabular reports, see the chapter titled “Reading Reports.”
- For descriptions of report writer variables and lists of the ID options you can use with each class of variable, see the chapter titled “Variables.”
- For descriptions of ID options and their use with commands and variables, see the chapter titled “ID Options.”

Syntax and operands for TAB reports are discussed in the following section. Sample output is also provided.

TABulate Command

Use the TAB command to produce tabular reports. The variables you specify with the TAB command determine the columns in your report and the order in which they appear. The operands you specify determine:

- How the report is organized: by date or time, by resource type or individual resources (such as transactions, terminals, or users), or by both
- How the report is sorted
- The column headings that appear in the report

Command Syntax and Operands

The syntax of the TAB command is as follows:

```
TAB [period] [idoption[(identifiers)]] variable [options] [HDR1(text)] [HDR2(text)] [SORT order] [FOR n] [AND,  
[TOP n]  
[BOTTOM n]  
variable [options] [HDR1(text)] [HDR2(text))] [AND,... ]
```

The operands for the TAB command have the following meanings:

Operand	Meaning
<i>period</i>	Sets the first column in the report to one of the following: DATETIME The date and time DATE The date TIME The time DAY The day (MONDAY, TUESDAY, and so on)
<i>idoption(identifiers)</i>	Any ID option that is valid for the variable, specified with or without identifiers. Specifying this operand causes the resulting report to be organized by the specified ID option and limited to the identifiers specified by <i>identifiers</i> . You can specify multiple ID options. Identifiers are listed either beginning in the first column (if you did not specify a time period as the first column) or beginning in the second column.
<i>variable</i>	Any report writer variable.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.
HDR1(<i>text</i>)	Overrides the default top heading for a column. (Each column has two default eight-character lines.) Replace <i>text</i> with the new column heading. To remove the header, replace <i>text</i> with a null string, as in HDR1().
HDR2(<i>text</i>)	Overrides the default bottom heading for a column. Replace <i>text</i> with the new column heading. To remove the header, replace <i>text</i> with a null string, as in HDR2().

Operand	Meaning
SORT <i>order</i>	<p data-bbox="716 321 1382 348">Sorts the report. Replace <i>order</i> with one of the following:</p> <p data-bbox="716 373 1382 527">DESCENDING – Variables are sorted in descending order starting with the variable the SORT operand is specified with, and, when the value is the same for two entries, from left to right. This is the default sorting order.</p> <p data-bbox="716 552 1382 674">ASCENDING – Variables are sorted in ascending order starting with the variable the SORT operand is specified with, and, when the value is the same for two entries, from left to right.</p> <p data-bbox="716 699 1382 793">ALPHA – Variables are sorted in alphanumeric order with respect to the specified identifier (for example, by transaction or terminal).</p> <p data-bbox="716 819 1382 905">You can use SORT only if the report is organized by identifier (for example, by transaction). If the report is organized by time, then it is already sorted.</p>
FOR <i>n</i>	<p data-bbox="716 930 1040 957">For Reports Organized by:</p> <p data-bbox="716 1003 821 1031">Identifier</p> <p data-bbox="716 1066 1032 1094">Identifier and Time or Date</p> <p data-bbox="716 1108 1382 1335">Limits reports to the <i>n</i> identifiers in each time that would appear first in the report, where <i>n</i> is an integer. For reports sorted in descending order, these are the <i>n</i> most active identifiers. For reports sorted in ascending order, these are the <i>n</i> least active identifiers. For reports sorted in alphanumeric order, these are the <i>n</i> identifiers coming first alphanumerically.</p> <p data-bbox="716 1360 1203 1388">For Reports Organized by Time or Date:</p> <p data-bbox="716 1402 1382 1633">Limits reports to the <i>n</i> times or dates with the highest values for the first variable specified. For reports sorted by a variable in descending order, these are the <i>n</i> times or dates with the highest value for that variable. For reports sorted by a variable in ascending order, these are the <i>n</i> times or dates with the lowest value for that variable.</p>

Operand	Meaning
TOP <i>n</i>	Limits reports organized by identifier to the <i>n</i> identifiers with the highest values for the first variable specified, where <i>n</i> is an integer. Unlike the FOR operand, which determines which identifiers to include in the report only after all the data has been read and processed, the TOP operand tests each record as it is read and discards records that do not qualify, saving storage. The TOP operand is appropriate for ID options with many unique identifiers, such as TRANNUM.
BOTTOM <i>n</i>	Identical to the TOP operand except it prints the <i>n</i> identifiers with the lowest values for the first variable specified.
AND,	Signals that another variable will be specified. The comma after AND lets you continue to the next line.

Organizing Reports by Time and Date or Resources

You can organize tabular reports by date or time, by a type of resource, or by both. Examples of each of these types of organization are given in the following sections.

Tabulating by Date and Time

Use the DATETIME operand to create tabular reports organized by date and time.

The following commands create a tabular report displaying transaction use, transaction lifetime, and maximum transaction lifetime. The DATETIME operand causes the report to be organized by date and time.

```
TAB DATETIME TRAN USE AND,
          TRAN LIFE AND,
          TRAN MAXLIFE
EACH HOUR
FROM 12/08/02 10:00:00
TO   12/08/02 18:00:00
END
RUN
```

The preceding commands generate a report like the following:

DATE	TIME	TRAN USE	TRAN LIFE	TRAN MAX LIFE
12/08/02	10.00	18	213.3	912.3
12/08/02	11.00	63	280.0	2980
12/08/02	12.00	39	515.7	5508
12/08/02	13.00	120	124.9	912.5
12/08/02	14.00	20	729.7	912.8
12/08/02	15.00	20	729.8	912.6
12/08/02	16.00	22	663.4	912.8
12/08/02	17.00	22	580.7	913.0
PERIOD		324	349.2	5508

Tabulating by Time

Use the TIME operand to create tabular reports organized by time.

In the preceding example, the date is the same for each row in the report. Such reports can be made briefer by omitting the date. To do so, use the TIME operand instead of the DATETIME operand, as follows:

```
TAB TIME TRAN USE AND,
      TRAN LIFE AND,
      TRAN MAXLIFE
EACH HOUR
FROM 12/08/02 10:00:00
TO 12/08/02 18:00:00
END
RUN
```

The preceding commands generate a report like the following:

TIME	TRAN USE	TRAN LIFE	TRAN MAX LIFE
10.00	18	213.3	912.3
11.00	63	280.0	2980
12.00	39	515.7	5508
13.00	120	124.9	912.5
14.00	20	729.7	912.8
15.00	20	729.8	912.6
16.00	22	663.4	912.8
17.00	22	580.7	913.0
PERIOD	324	349.2	5508

Tabulating by Date

Use the DATE operand to create tabular reports organized by date.

The following commands create a report in which data is organized by date:

```
TAB DATE TRAN USE AND,
      TRAN LIFE AND,
      TRAN MAXLIFE
END
RUN
```

The preceding commands generate a report like the following:

DATE	TRAN USE	TRAN LIFE	TRAN MAX LIFE
12/06/02	59	778.8	4529
12/07/02	1226	221.3	43995
12/08/02	518	638.9	46487
12/09/02	6881	16.13	4036
PERIOD	8684	87.43	46487

Tabulating by Day

Use the DAY operand to create tabular reports organized by the day of the week (MONDAY, TUESDAY, and so on).

The following commands create a report in which data is organized by day:

```
TAB DAY TRAN USE AND,
      TRAN ABENDS AND,
      TRAN CPU AND,
      TRAN LIFE AND,
      TRAN MAXLIFE
END
RUN
```

The preceding commands generate a report like the following:

WEEKDAY	TRAN USE	TRAN ABENDS	TRAN CPU	TRAN LIFE	TRAN MAX LIFE
FRIDAY	59	1	25.29	778.8	4529
SATURDAY	1226	60	0.208	221.3	43995
SUNDAY	518	3	0.392	638.9	46487
MONDAY	6881	77	0.515	16.13	4036
PERIOD	8684	141	0.633	87.43	46487

Tabulating by Resources

You can organize tabular reports by a type of resource, such as transactions or terminals, or by individual resources within a type. To do so, specify an ID option with or without identifiers following the TAB command and preceding any variables. Identifiers are listed in the first column, in descending order of the value of the first variable specified. For information about sorting reports in other ways, refer to *Sorting Reports* later in this chapter.

The following example uses the same commands used in the example used earlier in this chapter, except that the ID option TRANID replaces the DATETIME operand. This causes the resulting report to be organized by transaction rather than by date and time.

```
TAB TRANID TRAN USE      AND,
      TRAN LIFE  AND,
      TRAN MAXLIFE
EACH HOUR
FROM 12/08/02 10:00:00
TO   12/08/02 18:00:00
END
RUN
```

The preceding commands generate a report similar to the following:

TRAN ID	TRAN USE	TRAN LIFE	TRAN MAX LIFE
PSPI	97	0.149	1.060
TCP	57	912.2	913.0
KCP	57	912.2	913.0
POSS	40	0.092	0.593
CSPQ	27	0.032	0.557
CSNE	14	0.211	1.720
CATD	7	1.376	7.752
PINV	4	5.769	12.31
CSAC	4	0.039	0.085
RECE	3	137.1	379.4
JJJ	2	4244	5508
CSFU	2	79.80	151.7
CSSN	2	6.850	7.061
END	2	0.403	0.705
CSGM	2	0.029	0.045
VGBS	1	19.95	19.95
CRSQ	1	1.598	1.598
CSSF	1	0.767	0.767
VGBR	1	0.371	0.371
SUMMARY	324	349.2	5508

Tabulating by Both Time or Date and Resources

You can specify both an ID option and either the DATETIME, DATE, TIME, or DAY operand to create a more detailed tabular report.

The following commands generate a report organized by both time and transaction. The FOR operand limits the report to the five most active transactions during each interval.

```
TAB TIME TRANID FOR 5,
    TRAN USE AND,
    TERM RESP AND,
    DATASET USE
EACH HOUR
FROM 12/08/02 12:00:00
TO   12/08/02 15:00:00
END
RUN
```

The preceding commands generate a report like the following:

TIME	TRAN ID	TRAN USE	TERM RESP	FILE USE
12.00	PSPI	8	0.052	0
	TCP	8	0.000	0
	KCP	8	0.000	0
	POSS	5	0.058	3
	CSPQ	4	0.000	0
	SUMMARY	33	0.054	3
13.00	PSPI	84	0.046	0
	POSS	12	0.045	6
	KCP	8	0.000	0
	TCP	8	0.000	0
	CSPQ	4	0.000	0
	SUMMARY	116	0.046	6
14.00	TCP	8	0.000	0
	KCP	8	0.000	0
	CSPQ	4	0.000	0
	SUMMARY	20	0.000	0

Specifying Variables With or Without ID Options

You can specify multiple variables with the TAB command. The resulting report will include a column of data for each variable specified, in the order in which the variables are specified.

There are no restrictions on the variables you can include if you do not specify an ID option.

If you specify an ID option with the TAB command, the result is a report organized by the type of resource associated with the ID option. Accordingly, you can then use only variables that apply to that type of resource. For example, you cannot create reports showing transaction activity by dataset, program, or transient data. The variable classes that you can use with each ID option are listed in the table earlier in this chapter.

This does not mean, however, that you cannot use variables of different *classes* within a report organized by a type of resource. A report organized by terminal, for example, can include PROGRAM or TRANSACTION variables (among others), since these apply to terminal activity.

Using Both CICS and VSE Variables

If you are running both Unicenter CA-Explore for CICS and Unicenter CA-Explore for VSE, you can produce tabular reports containing data on both CICS and VSE. You can use any combination of CICS and VSE variables, subject to the restrictions described in the previous section.

To use both CICS and VSE variables, you must use the INPUT command to specify the Unicenter CA-Explore for CICS and the Unicenter CA-Explore for VSE files you want to use.

The following commands generate a report using data from both the Unicenter CA-Explore for VSE flashback file (EVSEFBK) and from the Unicenter CA-Explore for CICS flashback file (EXPCFBK):

```
INPUT(EVSEFBK,EXPCFBK)
TAB DATE VSE DISK SIO          HDR1(DISK) HDR2(SIO COUNT) AND,
      VSE DISK SIO RATE      HDR1(DISK) HDR2(SIO RATE) AND,
      VSE DISK SERVTIME          AND,
      VSE DISK MAXSERVTIME      AND,
      TRAN USE                   AND,
      TRAN LIFE
EACH 1 DAY
FROM 01/16/02
END
RUN
```

The preceding commands generate a report like the following:

DATE	DISK SIO COUNT	DISK SIO RATE	SERVICE TIME	MAX SERVTIME	TRAN USE	TRAN LIFETIME
01/16/02	2,156	0.02	0.023	0.093	154	1.218
01/17/02	170,270	1.97	0.023	0.244	461	0.822
01/18/02	74,139	2.42	0.018	0.069	223	0.761
PERIOD	246,565	0.53	0.022	0.244	838	0.878

Using ID Options with Variables

To limit the range of a variable to a specific resource or group of resources, specify an ID option and one or more identifiers following the variable.

Not all ID options can be used with all variables. For information about which ID options you can use with each variable class, refer to the description of the variable class in the chapter titled "Variables."

The following example uses the ID option JOBNAME to generate a single report containing information about two systems:

```

EACH 1 DAY
TAB DATE TRAN USE                HDR1(-TOTAL-) HDR2(TRAN USE)  AND,
      TERM RESP                  HDR1(-TOTAL-) HDR2(TERMRESP)  AND,
      TRAN USE  JOBNAME(CICS1)    HDR1(-CICS1-) HDR2(TRAN USE)  AND,
      TRAN USE  JOBNAME(CICS1) PCT HDR1(-CICS1-) HDR2(%USE)    AND,
      TERM RESP JOBNAME(CICS1)    HDR1(-CICS1-) HDR2(TERMRESP) AND,
      TRAN USE  JOBNAME(CICS2)    HDR1(-CICS2-) HDR2(TRAN USE)  AND,
      TRAN USE  JOBNAME(CICS2) PCT HDR1(-CICS2-) HDR2(%USE)    AND,
      TERM RESP JOBNAME(CICS2)    HDR1(-CICS2-) HDR2(TERMRESP)
END
RUN
    
```

The preceding commands generate a report similar to the following:

DATE TERMRESP	-TOTAL- TRAN USE	-TOTAL- TERMRESP	-CICS1- TRAN USE	-CICS1- %USE	-CICS1- TERMRESP	-CICS2- TRAN USE	-CICS2- %USE	-CICS2-
02/01/02	1134	2.342	761	67.1	1.445	307	27.1	4.844
02/02/02	2067	1.047	693	33.5	0.762	723	35.0	1.221
02/03/02	1893	1.758	730	38.6	1.942	463	24.5	2.008
02/04/02	1872	1.039	790	42.2	1.126	484	25.9	0.901
02/05/02	1061	0.772	665	62.7	0.794	143	13.5	1.066
02/06/02	1297	1.526	350	27.0	1.420	271	20.9	3.054
PERIOD	9324	1.602	3989	39.8	1.443	2391	24.7	2.280

Changing Column Headings

Each variable is assigned an abbreviation that is used by default as the heading for columns containing data about the variable. You can use the HDR1 and HDR2 operands to replace the top and bottom lines of the heading, respectively. There are no restrictions on the characters you can use in a heading.

Changing headings is especially useful if you:

- Change the type of data that a variable represents (for example, from a count to a rate).
- Use an ID option and identifiers to restrict the scope of a variable to particular resources, as shown in the example below.
- Need to shorten the column headings to gain space.

The variable TRAN USE is specified twice in the following example. By default, the column created for each occurrence of this variable would have the heading TRAN USE. Since the second occurrence of the variable applies only to the transaction PSPI, the HDR1 and HDR2 operands are included to create the heading TRAN USE FOR TRAN “PSPI” above the column for this variable.

```
TAB TIME TRAN USE      AND.
      TRAN LIFE      AND,
      TRAN MAXLIFE AND,
      TRAN USE TRANID(PSPI) HDR1( TRAN USE ) HDR2(FOR TRAN "PSPI")
EACH HOUR
FROM 12/08/02 12:00:00
TO   12/08/02 16:00:00
END
RUN
```

The preceding commands generate a report like the following:

TIME	TRAN USE	TRAN LIFE	TRAN MAX LIFE	TRAN USE FOR TRAN "PSPI"
12.00	39	515.7	5508	8
13.00	120	124.9	912.5	84
14.00	20	729.7	912.8	0
15.00	20	729.8	912.6	0
PERIOD	199	323.0	5508	92

Using Variable Options

You can use any of the variable options described in the chapter titled “Variables” with the TAB command.

The following options can be used only with the TAB command.

- DIFF
- RANGE
- RIGHT
- SELECT

Displaying Percentage of Change Over Time

In tabular reports organized by time or date, you can use the DIFF option to create a column displaying the percentage of change in a variable’s value between each time interval. The percentage of change is calculated with the following formula:

$$((\text{current value} - \text{previous value}) / \text{previous value}) * 100$$

Option Syntax

DIFF

The following commands use the DIFF option to display the percentage of change of transaction usage. Note that the HDR2 option assigns an appropriate column heading.

```
EACH 1 DAY
TAB DATE TRAN USE AND,
      TRAN USE DIFF HDR2 (%CHANGE)
END
RUN
```

The preceding commands generate a report like the following:

DATE	TRAN USE	TRAN %CHANGE
01/02/02	15,765	
01/03/02	22,165	+40.5%
01/04/02	66,374	+199.4%
01/05/02	91,302	+37.5%
01/06/02	2,535	-97.2%
01/07/02	2,123	-16.2%
PERIOD	200,264	

The difference between the number of transactions executed in the first interval and the second interval is 6400 (22,165 - 15,765), a 40.5% increase (6400 / 15,765 = 40.5%).

Changing a Count to a Rate

Use the RATE option with a variable that normally displays a count to display the rate for the variable.

The following commands produce a report that displays the transaction usage of all active transactions by transaction ID. The RATE option is used to give the transaction rate (number per second) as well.

```
EACH 1 DAY
TAB DATE TRAN USE      TRANID(CINQ) HDR1(CINQ) HDR2(USE)      AND,
      TRAN USE DIFF TRANID(CINQ) HDR1(CINQ) HDR2(%CHANGE) AND,
      TRAN USE RATE TRANID(CINQ) HDR1(CINQ) HDR2(RATE)  AND,
      TRAN USE      TRANID(PINQ) HDR1(PINQ) HDR2(USE)  AND,
      TRAN USE DIFF TRANID(PINQ) HDR1(PINQ) HDR2(%CHANGE) AND,
      TRAN USE RATE TRANID(PINQ) HDR1(PINQ) HDR2(RATE)

END
RUN
```

The preceding commands generate a report like the following:

DATE	CINQ USE	CINQ %CHANGE	CINQ RATE	PINQ USE	PINQ %CHANGE	PINQ RATE
01/02/02	2,978		0.03	12,778		0.15
01/03/02	5,304	+78.1%	0.06	16,861	+31.9%	0.20
01/04/02	30,734	+479.4%	0.36	35,452	+110.2%	0.41
01/05/02	38,711	+25.9%	0.45	52,322	+47.5%	0.61
01/06/02	2,113	-94.5%	0.02	395	-99.2%	0.00
01/07/02	417	-80.2%	0.01	1,610	+307.5%	0.02
PERIOD	80,257		0.16	119,418		0.24

The preceding report shows that in the 86,400 seconds on 01/02/92, transaction CINQ was executed 2978 times, which is a rate of 0.03 times per second.

Specifying a Time Period for a Variable

Use the SELECT option to apply to a particular variable the period specified with an INCLUDE or EXCLUDE command.

Option Syntax and Operand

SELECT (*name*)

Operand	Meaning
<i>name</i>	The <i>name</i> operand of the SELECT command must match the <i>name</i> operand of the ID operand for the INCLUDE or EXCLUDE command that specifies the time you want to apply to the variable.

The following example assigns the values of 1STSHIFT, 2NDSHIFT, and 3RDSHIFT to the *name* operands of the INCLUDE and SELECT commands. Each associated time period from the INCLUDE command is applied to the variable.

```
INCLUDE 08.00.00 - 15.59.59 ID 1STSHIFT
INCLUDE 16.00.00 - 23.59.59 ID 2NDSHIFT
INCLUDE 00.00.00 - 07.59.59 ID 3RDSHIFT
EACH 1 DAY
TAB DAY DATE TRAN USE          HDR1(TOTAL)   HDR2(USAGE)   AND,
          TRAN USE SELECT(1STSHIFT) HDR1(1ST SHIFT) HDR2(USAGE)   AND,
          TRAN USE SELECT(2NDSHIFT) HDR1(2ND SHIFT) HDR2(USAGE)   AND,
          TRAN USE SELECT(3RDSHIFT) HDR1(3RD SHIFT) HDR2(USAGE)   AND,
          TRAN LIFE          HDR1(AVG)     HDR2(LIFETIME) AND,
          TRAN LIFE SELECT(1STSHIFT) HDR1(1ST SHIFT) HDR2(LIFETIME) AND,
          TRAN LIFE SELECT(2NDSHIFT) HDR1(2ND SHIFT) HDR2(LIFETIME) AND,
          TRAN LIFE SELECT(3RDSHIFT) HDR1(3RD SHIFT) HDR2(LIFETIME)
END
RUN
```

The preceding commands generate a report like the following:

WEEKDAY	DATE	TOTAL USAGE	1ST SHIFT USAGE	2ND SHIFT USAGE	3RD SHIFT USAGE	AVG LIFETIME	1ST SHIFT LIFETIME	2ND SHIFT LIFETIME	3RD SHIFT LIFETIME
Friday	03/12/02	962	623	270	69	139.2	89.60	214.9	290.8
Saturday	03/13/02	417	199	18	200	288.7	290.3	270.3	288.8
Sunday	03/14/02	42		42		279.3		279.3	
Monday	03/15/02	365	175		190	287.4	272.9		300.8
PERIOD		1786	997	330	459	207.7	161.8	226.1	294.1

Shifting a Column to the Right

Use the RIGHT option to shift the column for a variable to the right. The entire column, including the headings, is shifted.

Option Syntax and Operand

RIGHT (*n*)

Operand	Meaning
<i>n</i>	The number of print columns (spaces) the column is to be shifted to the right

In the following example, the RIGHT variable option is specified with the variable TRAN USE to shift the column to the right so that it appears under the first heading specified with the HEADER command.

```

HEADER(20,24) -----TOTAL-----
HEADER(47,26) -----TRAN CSSN-----
TAB DAY TRAN USE RIGHT(8)    AND,
      TRAN LIFE              AND,
      TRAN CPU                AND,
      TRAN USE TRANID(CSSN) AND,
      TRAN LIFE TRANID(CSSN) AND,
      TRAN CPU  TRANID(CSSN)
END
RUN
    
```

The preceding commands generate a report like the following:

SHIFT NONE EACH DAY PERIOD NONE	-----TOTAL-----			-----TRAN CSSN-----		
	TRAN USE	TRAN LIFE	TRAN CPU	TRAN USE	TRAN LIFE	TRAN CPU
WEEKDAY						
Thursday	1258	205.1	0.032	129	8.756	0.010
Friday	369	175.3	0.028	66	1.773	0.011
PERIOD	1627	198.3	0.031	195	6.393	0.011

Limiting the Resources Included in a Report

You can limit a report to a subset of a resource type by specifying one or more identifiers with an ID option used with the TAB command.

The following commands include the ID option TERMID and the identifier 0A++ to display data about all 0A terminals:

```
TAB TERMID(0A++) TERM RESP AND,
          TERM I/OTIME AND,
          TRAN USE
END
RUN
```

The preceding commands generate a report like the following:

TERM ID	TERM RESP	TRAN TMIO	TRAN USE
0A0E	0.793	2.686	59
0A12	0.226	1.477	401
0A14	0.210	4.747	31
0A19	0.202	144.2	80
0A02	0.194	10.71	115
0A0F	0.172	34.89	6
0A11	0.158	0.719	13
0A18	0.103	0.002	6
0A08	0.082	1.970	23
0A0B	0.066	2.362	33
0A09	0.064	8.602	180
0A04	0.038	0.004	3
0A05	0.036	0.929	26
SUMMARY	0.189	15.96	976

Using the FOR Operand to Limit Reports Organized by Identifier

To limit the number of identifiers included in a report, use the FOR operand.

To insert a line in the report that represents all data excluded by the FOR operand, use the OPTION command and set the PRINT parameter to OTHERS.

You can add the operand FOR 5 to the commands used in the preceding example to limit the report to just those five 0A terminals with the highest response times, as follows:

```
TAB TERMID(0A++) FOR 5 TERM RESP AND,  
                        TERM I/OTIME AND,  
                        TRAN USE  
END  
RUN
```

The preceding commands generate a report like the following:

TERM ID	TERM RESP	TRAN TMIO	TRAN USE
0A0E	0.793	2.686	59
0A12	0.226	1.477	401
0A14	0.210	4.747	31
0A19	0.202	144.2	80
0A02	0.194	10.71	115
SUMMARY	0.220	19.92	686

Using the FOR Operand to Limit Reports Organized by Time and Date

To limit reports organized by time and date, use the FOR operand. The times included in the report are those with the highest value for the variable that the FOR operand is specified with.

The following commands produce a report limited to the five times with the highest values for the variable:

```
TAB DATETIME TRAN USE FOR 5
    EACH 1 HOUR
END
RUN
```

The preceding commands generate a report like the following:

DATE	TIME	TRAN USE
02/22/02	03.00	42
02/22/02	04.00	19
02/22/02	06.00	29
02/22/02	11.00	16
02/22/02	15.00	20
PERIOD		126

The following commands create a report on the 30 transactions with the shortest lifetimes. The FORMAT variable option is used to increase the number of digits in the output.

```
TAB TRANNUM TRAN LIFE FORMAT(N.NNNNNN) BOTTOM 30
END
RUN
```

The preceding commands generate a report like the following:

TRAN NUMBER	TRAN LIFE
0000080	0.0056320
0000037	0.0058880
0000775	0.0061440
0000045	0.0066560
0000236	0.0069120
0000890	0.0069120
0000111	0.0071680
0000248	0.0071680
0000786	0.0071680
0000947	0.0071680
0000118	0.0074240
0000247	0.0074240
0000251	0.0074240
0000308	0.0074240
0000425	0.0074240
0000893	0.0074240
0000142	0.0074240
0000144	0.0074240
0000233	0.0074240
0000242	0.0074240
0000549	0.0076800
0000682	0.0076800
0000229	0.0076800
0000401	0.0076800
0000379	0.0076800
0000139	0.0076800
0000136	0.0076800
0000141	0.0076800
0005510	0.0076800
0000102	0.0076800
SUMMARY	0.0071680

Sorting Reports

By default, tabular reports are sorted from left to right with respect to the variable values, in descending order. The leftmost variable is considered first; if the value is the same for two entries, the variable to the right is considered. This continues until variable values differ.

Consider the following commands and the report they generate:

```
TAB TERMID(0A++) FOR 5 TERM RESP AND,
                    TERM I/OTIME AND,
                    TRAN USE
END
RUN
```

The preceding commands generate a report like the following:

TERM ID	TERM RESP	TRAN TMIO	TRAN USE
0A0E	0.793	2.686	59
0A14	0.226	4.747	31
0A12	0.226	1.477	401
0A19	0.202	144.2	80
0A02	0.194	10.71	115
SUMMARY	0.220	19.92	686

In the preceding report, terminal 0A0E had the highest response time, so it is listed first. Terminals 0A14 and 0A12 have the same response time, but 0A14 had a higher I/O time, so it precedes 0A12 in the report.

Sorting Reports by Different Keys

To sort columns by different keys, you can alter the order of the columns.

To create a report similar to the preceding one but showing the five terminals using the most transactions rather than the five with the highest response times, move the variable TRAN USE so that it is the first variable in the TAB command, as follows:

```
TAB TERMID(0A++) FOR 5 TRAN USE      AND,
                        TERM RESP    AND,
                        TERM I/OTIME
END
RUN
```

The preceding commands generate a report like the following:

TERM ID	TRAN USE	TERM RESP	TRAN TMIO
0A12	401	0.226	1.477
0A09	180	0.064	8.602
0A02	115	0.194	10.71
0A19	80	0.202	144.2
0A0E	59	0.793	2.686
SUMMARY	835	0.197	18.05

Sorting Reports in Ascending Order

To sort reports in ascending order, add the SORT ASCENDING operand to the TAB command.

The following commands produce a report sorted so that the five terminals using the fewest transactions are listed in ascending order of transaction use:

```
TAB TERMID(0A++) FOR 5 SORT ASCENDING,
    TRAN USE AND,
    TERM RESP AND,
    TERM I/OTIME
END
RUN
```

The preceding commands generate a report like the following:

TERM ID	TRAN USE	TERM RESP	TRAN TMIO
0A04	3	0.038	0.004
0A18	6	0.103	0.002
0A0F	6	0.172	34.89
0A11	13	0.158	0.719
0A08	23	0.082	1.970
SUMMARY	51	0.109	5.177

You can specify the SORT operand with a variable other than the first variable. For example, the following commands generate a report sorted by transaction use, the third variable specified:

```
TAB TRANID TRAN TCPU AND,
      FILE USE AND,
      TRAN USE SORT ASCENDING
END
RUN
```

The preceding commands generate a report like the following:

TRAN ID	TRAN CPU	FILE USE	TRAN USE
RAPP	0.098	0	2
PINV	1.486	26	2
PADI	0.350	0	2
SPAM	0.132	3	3
RAPL	0.011	0	4
CSSN	0.071	0	8
CSGM	0.007	0	8
CATD	0.105	0	9
JJJ	0.221	0	10
CSPQ	0.113	0	10
VGBC	0.055	0	13
POSS	0.035	21	37
PSPI	0.097	0	43
KCP	7.581	0	45
TCP	0.572	241	46
CSNE	0.010	0	49
X25T	0.132	198	57
SUMMARY	1.123	489	348

Sorting Reports by Identifier

To sort reports by identifier, add the SORT ALPHA operand to the TAB command.

Example

The following commands generate a report sorted alphanumerically by transaction ID:

```
TAB TRANID FOR 20 SORT ALPHA,
TRAN LIFE AND TRAN USE AND TRAN MAXLIFE
END
RUN
```

The preceding commands generate a report like the following:

TRAN ID	TRAN LIFE	TRAN USE	TRAN MAX LIFE
BOAG	14.22	4	27.86
BOEF	35.32	2	58.14
BOEN	44.51	8	130.6
CATD	0.465	125	7.752
CEMT	60.92	4	218.6
CHAI	0.193	160	1.475
CHAT	0.190	24	0.802
CIRB	9.981	2	13.30
CRSQ	0.795	7	3.738
CSAC	0.066	25	0.647
CSDA	0.510	2	1.004
CSDB	45241	2	46487
CSDC	3957	3	4529
CSDD	0.192	1	0.192
CSFU	79.80	2	151.7
CSGM	1.030	107	56.41
CSKP	1.645	6	4.393
CSMT	13.30	2	26.08
CSNE	0.026	699	1.720
CSPG	0.032	157	0.533
SUMMARY	77.15	1342	0.000

Using Other Commands with the TAB Command

You can use the secondary commands to tailor tabular reports. Examples of the use of the following commands in tabular reports are given in the following sections:

- RANGE
- GROUP
- COST

Tabulating by Ranges

You can use either the RANGE command or the RANGE option to restrict a report to resources whose activity falls within a specified range. Examples of its use in tabular reports are given in the following sections.

The RANGE command can be used to test range criteria either at the record level (immediate) or, by specifying the DEFERRED operand, just prior to printing the report (deferred).

Immediate range criteria are tested before the record is processed. If the range criteria are not met, the record is discarded for that particular report.

Deferred range criteria are not tested until all data is read. The range criteria are checked just before each row is printed, and if any variable in the row fails to meet its range criteria, the entire row is discarded.

Immediate Range Criteria

The following commands produce a report about terminals. The RANGE command is specified without the DEFERRED operand to exclude from the report any individual record for which the value of TERM INPUTS is greater than 50.

```
TAB TERMID(0A++),  
    TERM INPUTS AND,  
    TERM RESP AND,  
    TERM I/OTIME AND,  
    TRAN USE  
RANGE TERM INPUTS 0-50  
END  
RUN
```

The preceding commands generate a report like the following:

TERM ID	TERM READS	TERM RESP	TRAN TMIO	TRAN USE
0A12	401	0.226	1.477	401
0A19	188	0.107	48.08	76
0A09	166	0.064	8.602	180
0A02	143	0.216	4.757	114
0A0E	57	0.793	2.686	59
0A0B	37	0.066	2.362	33
0A14	33	0.210	4.747	31
0A05	25	0.036	0.929	26
0A08	23	0.082	1.970	23
0A11	13	0.158	0.719	13
0A0F	10	0.172	34.89	6
0A18	5	0.103	0.002	6
0A04	2	0.038	0.004	3
SUMMARY	1103	0.168	7.219	971

Deferred Range Criteria

Note that in the preceding example, several terminals with a total number of inputs ("TERM READS") not in the specified range are included in the report. Because the RANGE command was specified without the DEFERRED operand, the test of the number of inputs was applied to each individual record, not to the total number of inputs for each terminal. Although individual records with more than 50 inputs were excluded, the total for a given terminal can exceed the range.

If instead of the preceding report, you wanted a report on the terminals that had fewer than 50 total inputs, you would use the same commands as those in the previous example, but specify the DEFERRED operand with the RANGE command. This causes the report writer to apply the range criteria only after all the input data is read, enabling it to determine whether each terminal had fewer than 50 total inputs.

```
TAB TERMID(0A++),
    TERM INPUTS AND,
    TERM RESP AND,
    TERM I/OTIME AND,
    TRAN USE
RANGE TERM INPUTS 0-50 DEFERRED
END
RUN
```

The preceding commands generate a report like the following:

TERM ID	TERM READS	TERM RESP	TRAN TMIO	TRAN USE
0A0B	37	0.066	2.362	33
0A14	33	0.210	4.747	31
0A05	25	0.036	0.929	26
0A08	23	0.082	1.970	23
0A11	13	0.158	0.719	13
0A0F	10	0.172	34.89	6
0A18	5	0.103	0.002	6
0A04	2	0.038	0.004	3
SUMMARY	148	0.107	3.640	141

Using the RANGE Option

Using the RANGE option yields the same results as using the RANGE command with the DEFERRED operand. For example, the following commands generate the same report as the preceding commands but include the RANGE option instead of the RANGE command:

```
TAB TERMID(0A++),
    TERM INPUTS RANGE(0,50) AND,
    TERM RESP AND,
    TERM I/OTIME AND,
    TRAN USE
END
RUN
```

Tabulating by Groups

Use the GROUP command to define groups of resources you want a report to be about.

You can also include groups in a report by using the OPTION command and setting the PRINT parameter to GROUP.

The following example uses the GROUP command to define groups by terminal ID. In this example, four divisions are defined. Since each group name begins with CO, the groups can be selected in the tabular report by specifying TERMID(CO*).

```
GROUP TERMID CO.NORTHERN.DIVISION(L7*)
GROUP TERMID CO.SOUTHERN.DIVISION(L4*)
GROUP TERMID CO.EASTERN.DIVISION(L2*)
GROUP TERMID CO.WESTERN.DIVISION(L1*)

TAB TERMID(CO*) HDR1(DEPARTMENT) HDR2(NAME),
    TRAN USE AND,
    TRAN USE RATE HDR2(RATE) AND,
    TRAN LIFE AND,
    TRAN CPU AND,
    TERM RESP AND,
    FILE USE

END
RUN
```

The preceding commands generate a report like the following:

DEPARTMENT NAME	TRAN USE	TRAN RATE	TRAN LIFE	TRAN CPU	TERM RESP	FILE USE
CO.WESTERN.DIVISION	3472	0.02	4.599	0.094	0.360	16925
CO.NORTHERN.DIVISION	976	0.01	16.52	0.120	0.189	1080
CO.SOUTHERN.DIVISION	728	0.00	4.770	0.073	0.106	289
CO.EASTERN.DIVISION	667	0.00	3.688	0.180	0.637	8032
SUMMARY	5843	0.03	6.508	0.106	0.290	26326

Tabulating by Both Group and Range

You can use the GROUP and RANGE commands together to produce a report on groups of resources whose activity falls within different ranges.

The following example uses both the GROUP and RANGE commands to define 13 groups of transactions in terms of their lifetimes:

```
GROUP TRANID RANGE:00.000-01.000(*) RANGE(TLIFE1)
GROUP TRANID RANGE:01.001-02.000(*) RANGE(TLIFE2)
GROUP TRANID RANGE:02.001-03.000(*) RANGE(TLIFE3)
GROUP TRANID RANGE:03.001-04.000(*) RANGE(TLIFE4)
GROUP TRANID RANGE:04.001-05.000(*) RANGE(TLIFE5)
GROUP TRANID RANGE:05.001-06.000(*) RANGE(TLIFE6)
GROUP TRANID RANGE:06.001-07.000(*) RANGE(TLIFE7)
GROUP TRANID RANGE:07.001-08.000(*) RANGE(TLIFE8)
GROUP TRANID RANGE:08.001-09.000(*) RANGE(TLIFE9)
GROUP TRANID RANGE:09.001-10.000(*) RANGE(TLIFE10)
GROUP TRANID RANGE:10.001-20.000(*) RANGE(TLIFE20)
GROUP TRANID RANGE:20.001-50.000(*) RANGE(TLIFE50)
GROUP TRANID RANGE:50.001+(*) RANGE(TLIFE50+)
```

```
RANGE TRAN LIFE 0-1 ID TLIFE1
RANGE TRAN LIFE 1-2 ID TLIFE2
RANGE TRAN LIFE 2-3 ID TLIFE3
RANGE TRAN LIFE 3-4 ID TLIFE4
RANGE TRAN LIFE 4-5 ID TLIFE5
RANGE TRAN LIFE 5-6 ID TLIFE6
RANGE TRAN LIFE 6-7 ID TLIFE7
RANGE TRAN LIFE 7-8 ID TLIFE8
RANGE TRAN LIFE 8-9 ID TLIFE9
RANGE TRAN LIFE 9-10 ID TLIFE10
RANGE TRAN LIFE 10-20 ID TLIFE20
RANGE TRAN LIFE 20-50 ID TLIFE50
RANGE TRAN LIFE 50-99999 ID TLIFE50+
```

```
TAB TRANID(RANGE*) HDR1(TRAN LIFETIME) HDR2(RANGES),
  SORT ALPHA,
```

```
TRAN USE AND,
TRAN USE PCT AND,
TRAN LIFE AND,
TRAN MAXLIFE AND,
TERM RESP AND,
TRAN CPU AND,
TRAN I/OTIME AND,
TRAN USERTIME AND,
TRAN CICSTIME
```

```
END
RUN
```

The preceding commands generate a report like the following:

TRAN LIFETIME RANGES	TRAN USE	TRAN % USE	TRAN LIFE	TRAN MAX LIFE	TERM RESP	TRAN CPU	TRAN I/O	TRAN USER	TRAN CICS
RANGE:00.000-01.000	6981	80.4	0.207	0.999	0.170	0.052	0.053	0.000	0.000
RANGE:01.001-02.000	315	3.6	1.469	1.998	0.726	0.269	0.213	0.000	0.000
RANGE:02.001-03.000	140	1.6	2.502	2.993	1.006	0.356	0.200	0.000	0.000
RANGE:03.001-04.000	92	1.1	3.486	3.989	1.134	0.397	0.199	0.000	0.000
RANGE:04.001-05.000	55	0.6	4.417	4.928	1.380	0.499	0.367	0.000	0.000
RANGE:05.001-06.000	41	0.5	5.499	5.975	1.082	0.434	0.294	0.000	0.000
RANGE:06.001-07.000	40	0.5	6.401	6.847	1.295	0.377	0.092	0.000	0.000
RANGE:07.001-08.000	41	0.5	7.496	7.990	0.741	0.278	0.472	0.000	0.000
RANGE:08.001-09.000	12	0.1	8.545	8.950	0.700	0.419	0.074	0.000	0.000
RANGE:09.001-10.000	18	0.2	9.462	9.924	0.687	0.254	0.273	0.000	0.000
RANGE:10.001-20.000	217	2.5	15.80	19.95	0.445	0.232	0.055	0.000	0.000
RANGE:20.001-50.000	99	1.1	31.62	49.13	0.611	0.545	0.374	0.000	0.000
RANGE:50.001+	633	7.3	1182	46487	0.259	7.542	0.243	0.000	0.000
SUMMARY	8684	100.0	87.43	46487	0.275	0.633	0.086	0.000	0.000

Creating Charge-Back Reports

Use the COST command with the TAB command to create charge-back reports.

For information about creating distributed cost reports, refer to the description of the COST command .

The COST commands in the following example cause one cent to be charged for every transaction, 25 cents for every CPU second used, 1/10 cent per file request, and 1/2 cent per file add. The TAB command makes use of these cost factors, creating a tabular report with a column showing the value of each variable for each transaction. The HDR1 and HDR2 operands assign those columns appropriate headings. The last variable, REPORT COST TOTAL, creates a column that shows the sum of the costs for each row in the report.

```

COST TRAN USE .01
COST TRAN CPU .25
COST FILE USE .001
COST FILE ADDS .005

TAB TRANID TRAN USE AND,
          TRAN USE COST HDR1(TRAN USE) HDR2(COST) AND,
          TRAN CPU AND,
          TRAN CPU COST HDR1(TRAN CPU) HDR2(COST) AND,
          FILE USE AND,
          FILE USE COST HDR1(FILE USE) HDR2(COST) AND,
          FILE ADDS AND,
          FILE ADDS COST HDR1(FILE ADDS) HDR2(COST) AND,
          REPORT COST TOTAL

END
RUN

```

The preceding commands generate a report like the following:

TRAN ID	TRAN USE	TRAN USE COST	TRAN CPU	TRAN CPU COST	FILE USE	FILE USE COST	FILE ADDS	FILE ADDS COST	TOTAL COST
X25T	57	\$0.57	0.132	\$1.88	198	\$0.20	9	\$0.05	\$2.70
CSNE	49	\$0.49	0.010	\$0.12	0	\$0.00	0	\$0.00	\$0.61
TCP	46	\$0.46	0.572	\$6.57	241	\$0.24	0	\$0.00	\$7.27
KCP	45	\$0.45	7.581	\$85.28	0	\$0.00	0	\$0.00	\$85.73
PSPI	43	\$0.43	0.097	\$1.04	0	\$0.00	0	\$0.00	\$1.47
POSS	37	\$0.37	0.035	\$0.32	21	\$0.02	0	\$0.00	\$0.71
VGBC	13	\$0.13	0.055	\$0.18	0	\$0.00	0	\$0.00	\$0.31
JJJ	10	\$0.10	0.221	\$0.55	0	\$0.00	0	\$0.00	\$0.65
CSPQ	10	\$0.10	0.113	\$0.28	0	\$0.00	0	\$0.00	\$0.38
CATD	9	\$0.09	0.105	\$0.24	0	\$0.00	0	\$0.00	\$0.33
CSSN	8	\$0.08	0.071	\$0.14	0	\$0.00	0	\$0.00	\$0.22
CSGM	8	\$0.08	0.007	\$0.01	0	\$0.00	0	\$0.00	\$0.09
RAPL	4	\$0.04	0.011	\$0.01	0	\$0.00	0	\$0.00	\$0.05
SPAM	3	\$0.03	0.132	\$0.10	3	\$0.00	0	\$0.00	\$0.13
PINV	2	\$0.02	1.486	\$0.74	26	\$0.03	0	\$0.00	\$0.79
PADI	2	\$0.02	0.350	\$0.17	0	\$0.00	0	\$0.00	\$0.19
RAPP	2	\$0.02	0.098	\$0.05	0	\$0.00	0	\$0.00	\$0.07
SUMMARY	348	\$3.48	1.123	\$195.41	489	\$0.49	9	\$0.05	\$199.43

Charge-Back Reports Organized by Group

Use the COST command with the GROUP command to create charge-back reports organized by groups. The following commands use terminal IDs to define company divisions and list their corresponding charges in the report:

```

GROUP TERMID CO.NORTHERN.DIVISION(L4*)
GROUP TERMID CO.SOUTHERN.DIVISION(L3*)
GROUP TERMID CO.EASTERN.DIVISION(L2*)
GROUP TERMID CO.WESTERN.DIVISION(L1*)

COST TRAN USE .01
COST TRAN CPU .25
COST FILE USE .001
COST FILE ADDS .005

TAB TERMID(CO*) HDR1(DEPARTMENT) HDR2(NAME),
  TRAN USE AND,
  TRAN USE COST HDR2(COST) AND,
  TRAN CPU COST HDR1(CPU) HDR2(COST) AND,
  FILE USE COST HDR1(FILE USE) HDR2(COST) AND,
  FILE ADD COST HDR1(FILE ADD) HDR2(COST) AND,
  REPORT COST TOTAL

END
RUN

```

The preceding commands generate a report like the following:

DEPARTMENT NAME	TRAN USE	TRAN COST	CPU COST	FILE USE COST	FILE ADD COST	TOTAL COST
CO. WESTERN. DIVISION	4415	\$44.15	\$32.29	\$117.61	\$0.98	\$195.03
CO. NORTHERN. DIVISION	2224	\$22.24	\$15.15	\$38.52	\$0.16	\$76.08
CO. EASTERN. DIVISION	2035	\$20.35	\$15.79	\$59.13	\$0.65	\$95.92
CO. SOUTHERN. DIVISION	2024	\$20.24	\$43.73	\$288.63	\$0.49	\$353.09
SUMMARY	10698	\$106.98	\$106.97	\$503.91	\$2.28	\$720.13

TAB2 Reports

Examples of the use of the following command and operands used in TAB2 reports are given in the following sections:

- TAB2 command
- FOR operand
- TOP or BOTTOM operand

TAB2 Command

Use the TAB2 command to present more information about each resource reported upon by the TAB command.

Option Syntax and Operands

The syntax of the TAB2 command is as follows:

```
TAB2 idoption[(identifiers)] variable [options] [HDR1(text)] [HDR2(text)] [SORT order] [FOR n] [AND,
[TOP n]
[BOTTOM n]
variable [options] [HDR1(text)] [HDR2(text))] [AND,...]
```

The TAB2 command uses the same operands as the TAB command, except that you cannot specify both a period operand (DATETIME, DATE, TIME, or DAY) and an ID option. For an explanation of the TAB2 operands, refer to the description of the TAB operands.

You cannot use the TAB2 command without the TAB command.

You can use the TAB2 command only if you also specify the TAB command with an ID option. You can specify a period option (DATETIME, DATE, TIME, and DAY) with either the TAB command or the TAB2 command, but **not** with both.

Using the FOR Operand to Limit TAB2 Reports

To limit the number of identifiers included in a report for each identifier listed by the TAB command, use the FOR operand with the TAB2 command.

The following TAB and TAB2 commands create a report showing terminals and the five transactions used most by each:

```
TAB  TERMID      TRAN USE  AND,
      TRAN LIFE  AND,
      TRAN CPU
TAB2  TRANID FOR 5 TRAN USE  AND,
      TRAN LIFE  AND,
      TRAN CPU
END
RUN
```

The preceding commands generate a report like the following, shown in part:

TERM ID	TRAN USE	TRAN LIFE	TRAN CPU	TRAN ID	TRAN USE	TRAN LIFE	TRAN CPU
WA19	65	1.943	0.037	CSPG	40	0.033	0.002
				GLAE	23	5.193	0.098
				CSSN	1	5.527	0.080
				CSGM	1	0.037	0.002
				SUMMARY	65	1.943	0.037
4A93	53	18.37	0.116	WISB	51	19.01	0.121
				CSSN	1	4.460	0.019
				CSGM	1	0.012	0.002
				SUMMARY	53	18.37	0.116
BP3P 0A0B	42 33	2.921 2.424	0.088 0.020	RAPP	42	2.921	0.088
				CSPG	18	0.022	0.002
				PADI	14	5.683	0.044
				CSGM	1	0.017	0.002
				SUMMARY	33	2.424	0.020
0A14	31	5.125	0.086	WISI	11	14.26	0.199
				CHAI	11	0.071	0.022
				PART	7	0.173	0.035
				CSGM	2	0.015	0.002
				SUMMARY	31	5.125	0.086
SUMMARY	8684	87.43	0.633		8684	87.43	0.633

The following example combines the FOR operand with both the TAB and TAB2 commands to create a table of transactions and the terminals associated with them:

```
TAB  TRANID FOR 3 TRAN USE  AND,
      TRAN LIFE AND,
      TRAN CPU
TAB2 TERMID FOR 9 TRAN USE  AND,
      TRAN LIFE AND,
      TRAN CPU

END
RUN
```

The preceding commands generate a report like the following:

TRAN ID	TRAN USE	TRAN LIFE	TRAN CPU	TERM ID	TRAN USE	TRAN LIFE	TRAN CPU
FXFR	2482	0.361	0.072	F014	2482	0.361	0.072
PSPI	939	0.248	0.092	8A21	197	0.219	0.084
				F006	116	0.327	0.158
				0A09	67	0.198	0.103
				0A02	64	0.224	0.103
				8A03	60	0.228	0.060
				8A09	57	0.163	0.048
				F012	41	0.271	0.106
				4A64	41	0.235	0.079
				8A24	40	0.370	0.102
				SUMMARY	683	1305	517.8
CSNE	699	0.026	0.007	N/A	699	0.026	0.007
SUMMARY	4120	0.279	0.066		8684	87.43	0.633

Using the TOP or BOTTOM Operand to Limit TAB2 Reports

The TOP and BOTTOM operands, like the FOR operand, limit reports organized by identifier to the number of identifiers specified. However, while the FOR operand tests identifiers only after all the data has been read and processed, the TOP and BOTTOM operands cause the report writer to test each record as it is read and discard records that do not qualify, saving storage. The TOP and BOTTOM operands are particularly appropriate for ID options with many unique identifiers, such as TRANNUM.

The TAB command in the following example creates a report on the terminal response time of transactions. The TAB2 command shows the four transactions that had the longest lifetimes for each terminal. Processing similar commands using the FOR operand instead of the TOP operand would require much more storage.

```
TAB  TERMID TERM RESP
TAB2 TRANNUM TRAN LIFE TOP 4
END
RUN
```

The preceding commands generate a report like the following, shown in part:

TERM ID	TERM RESP	TRAN NUMBER	TRAN LIFE
CNSL	2.149	0000020	13.30
		0000019	0.016
		SUMMARY	6.663
ZA00	0.278	0000054	306.6
		0000058	0.647
		0000059	0.486
		0000053	0.264
		SUMMARY	77.01
0A19	0.202	0000297	2390
		0000613	2283
		0000427	2209
		0000229	1600
		SUMMARY	2121
.	.	.	.
.	.	.	.
SUMMARY	0.275		4176

Plot Reports

Plot reports plot the value of a variable for a group of resources either against time or against another variable. This chapter describes and gives examples of how to generate plot reports using the following commands: PLOT, PLOT2, VPLOT, VPLOT2, MPLOT, HPLOT.

Plot reports fall into three categories:

- **Horizontal plot reports** – Are generated by the PLOT and PLOT2 commands, plot a variable horizontally across time.
- **Vertical plot reports** – Are generated by the VPLOT, VPLOT2, and MPLOT commands, plot a variable vertically across time.
- **Distribution plot reports** – Are generated by the HPLOT command used with either the VPLOT or the MPLOT command, plot one variable on the vertical axis and another on the horizontal axis.

For more information on related topics, refer to the following chapters:

- See the chapter titled “Commands” to become familiar with report writer commands.
- For an explanation of the structure of each kind of plot report, see the chapter titled “Reading Reports.”
- For descriptions of report writer variables and lists of the ID options you can use with each class of variable, see the chapter titled “Variables.”
- For descriptions of ID options and their use with commands and variables, see the chapter titled “ID Options.”

Using Both CICS and VSE Variables

If you are running both Unicenter CA-Explore for CICS and Unicenter CA-Explore for VSE, you can produce plot reports containing data on both CICS and VSE. You can use any combination of CICS and VSE variables.

To use both CICS and VSE variables, you must use the INPUT command to specify the Unicenter CA-Explore for CICS and the Unicenter CA-Explore for VSE files you want to use. For more information about INPUT command, see the chapter titled “Commands.” For descriptions of Unicenter CA-Explore for VSE variables, refer to the Unicenter CA-Explore for VSE *History Reporting Guide*.

Horizontal Reports (PLOT and PLOT2)

This section discusses operands and sample commands, and provides sample output for the horizontal reports generated by the PLOT and PLOT 2 commands.

PLOT Command

Use the PLOT command to produce reports that plot statistics about resources for one or more times.

Command Syntax and Operands

The syntax of the PLOT command is as follows:

```
PLOT variable [options] [ALONE] [SCALE({n},{mX})] [CHAR1(x)]
                                     [CHAR2(x)]
```

The operands for the PLOT command have the following meanings:

Operand	Meaning
<i>variable</i>	Any report writer variable.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.
ALONE	Prevents the display of the global (system-wide) value of the variable. To use this operand, you must restrict the variable with an ID option.
SCALE(<i>n</i>) or SCALE(<i>mX</i>) or SCALE(<i>n,mX</i>)	By default, the highest value of a plot's horizontal axis equals the highest value reached by a variable in the plot or, for variables representing percentages, 100%. To override this dynamic scale, specify <i>n</i> as the maximum value. If the variable that you choose produces integers, <i>n</i> must be an integer. In some cases, bar graphs in a plot report will be too short to appear. To display these graphs, you can cause each plot character to appear as <i>m</i> characters by specifying the multiplier <i>mX</i> , where <i>m</i> is an integer. Since the resulting bar graphs will no longer correspond numerically to the scale, this is useful only for determining relative performance.
CHAR1(<i>x</i>)	Sets the plot character that represents the restricted variable. Replace <i>x</i> with any valid EBCDIC character.
CHAR2(<i>x</i>)	Sets the plot character that represents the global variable. Replace <i>x</i> with any valid EBCDIC character.

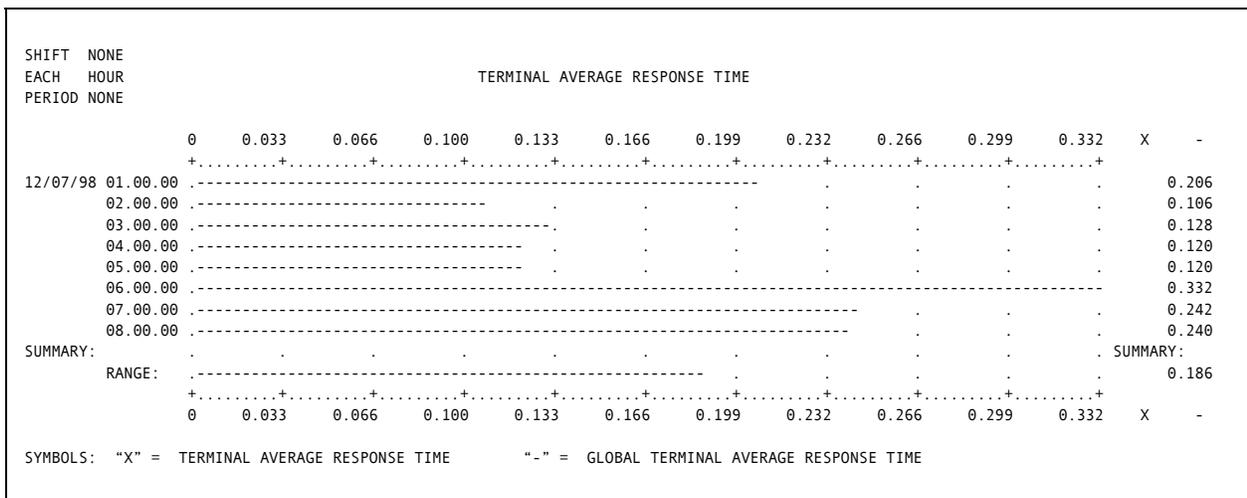
Creating a Simple PLOT Report

To create a simple plot report, use the PLOT command followed by a variable to be plotted.

For example, the following commands produce a plot of average terminal response time, with each detail line representing an hour, for the morning of a single day:

```
PLOT TERM RESPONSETIME
  FROM 12/07/98 01:00:00
  TO   12/07/98 09:00:00
  EACH HOUR
END
RUN
```

The preceding commands generate a report like the following:



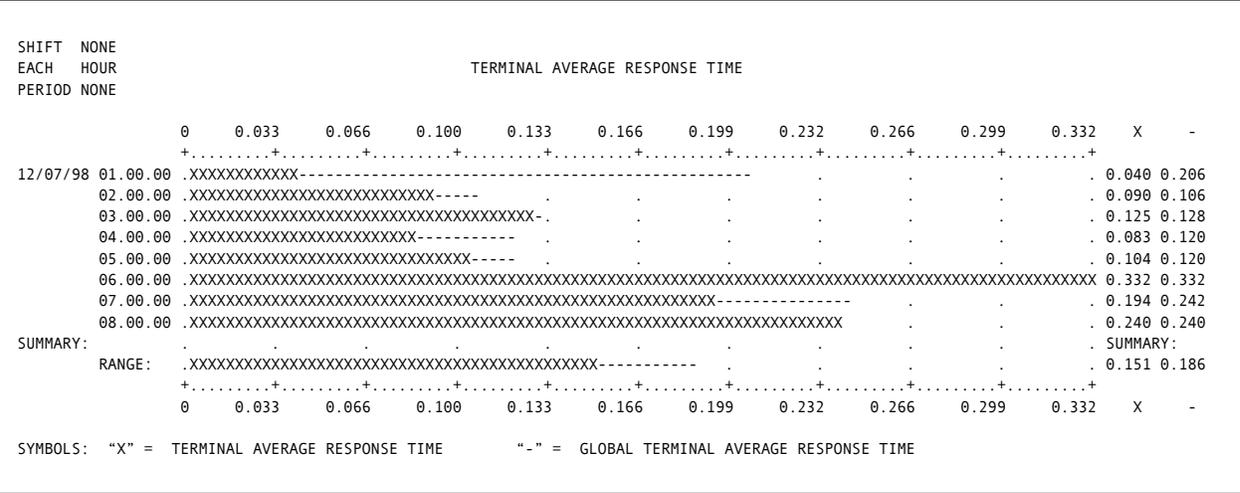
Using ID Options in Plot Reports

To plot the performance of one or more individual resources (such as transactions, terminals, or operators), follow the variables to be plotted with an ID option, and a list of one or more identifiers (such as terminal IDs or transaction IDs). You can use generic characters to specify a group of identifiers.

The following PLOT command includes the variable TERM RESP, followed by the ID option TERMD, and the identifier 0A to produce a plot of the average response time for terminals having IDs beginning with the characters 0A:

```
PLOT TERM RESP TERMD(0A*)
FROM 12/07/98 01:00:00
TO 12/07/98 09:00:00
EACH HOUR
END
RUN
```

The preceding commands generate a report like the following. Along with the average response time of 0A terminals (plotted with Xs), the average response time for all terminals is also plotted with hyphens, enabling you to easily compare the two.



Excluding Global Values from a Plot

Use the ALONE operand to plot the activity of one or more identifiers without including a plot of global activity in the report.

Suppose you want to plot the response time for 0A terminals, but you do not want to plot the average response time for all terminals. You can use the same commands as in the previous example but add the ALONE operand, as follows:

```
PLOT TERM RESP TERMID(0A*) ALONE
FROM 12/07/98 01:00:00
TO 12/07/98 09:00:00
EACH HOUR
END
RUN
```

The preceding commands generate a report like the following:

SHIFT	NONE	TERMINAL AVERAGE RESPONSE TIME											
EACH	HOUR	0	0.033	0.066	0.100	0.133	0.166	0.199	0.232	0.266	0.299	0.332	X
PERIOD	NONE												
12/07/98	01.00.00	.XXXXXXXXXXXXX 0.040
	02.00.00	.XXXXXXXXXXXXXXXXXXXXXXXXXXXX 0.090
	03.00.00	.XXXXXXXXXXXXXXXXXXXXXXXXXXXX 0.125
	04.00.00	.XXXXXXXXXXXXXXXXXXXXXXXXXXXX 0.083
	05.00.00	.XXXXXXXXXXXXXXXXXXXXXXXXXXXX 0.104
	06.00.00	.XXXXXXXXXXXXXXXXXXXXXXXXXXXX 0.332
	07.00.00	.XXXXXXXXXXXXXXXXXXXXXXXXXXXX 0.194
	08.00.00	.XXXXXXXXXXXXXXXXXXXXXXXXXXXX 0.240
SUMMARY:													SUMMARY:
	RANGE:	.XXXXXXXXXXXXXXXXXXXXXXXXXXXX 0.151
		+.....+.....+.....+.....+.....+.....+.....+.....+.....+.....+.....+											
		0	0.033	0.066	0.100	0.133	0.166	0.199	0.232	0.266	0.299	0.332	X

SYMBOLS: "X" = TERMINAL AVERAGE RESPONSE TIME

Changing a Plot's Scale

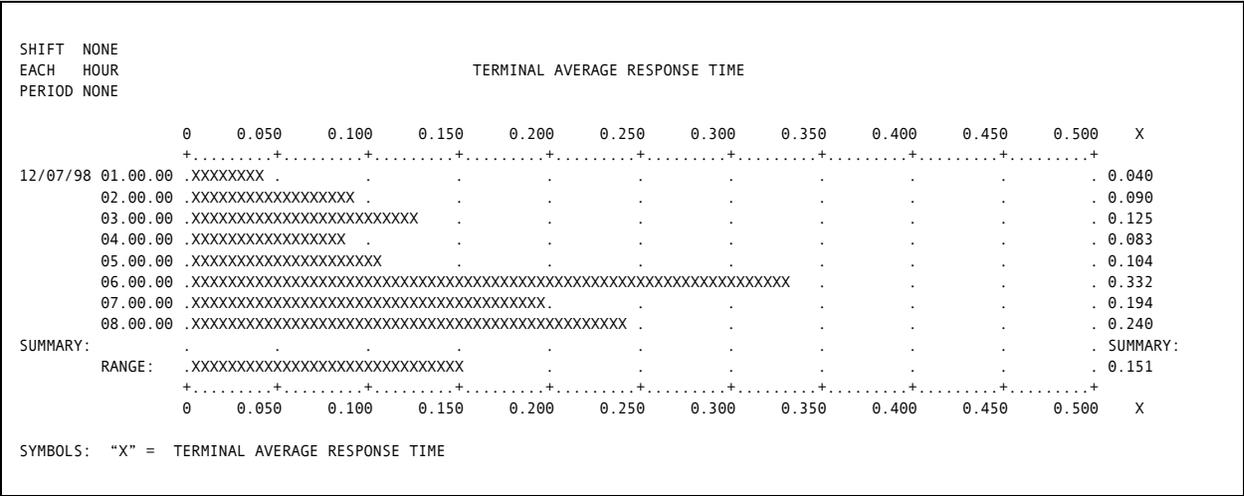
By default, the highest value of a plot's scale equals the highest value reached by a variable in the plot or, for variables representing percentages, 100%. Use the SCALE operand to set your own scale.

If the variable you are plotting yields integer values, you can specify only an integer value with the SCALE operand. If the variable yields real values, the scale you specify must include a decimal point.

To change the maximum value of the scale used in the preceding example to 0.500 seconds, add the SCALE(0.500) operand, as follows:

```
PLOT TERM RESP TERMID(0A*) ALONE SCALE(0.500)
FROM 12/07/98 01:00:00
TO 12/07/98 09:00:00
EACH HOUR
END
RUN
```

These commands generate a report like the following:



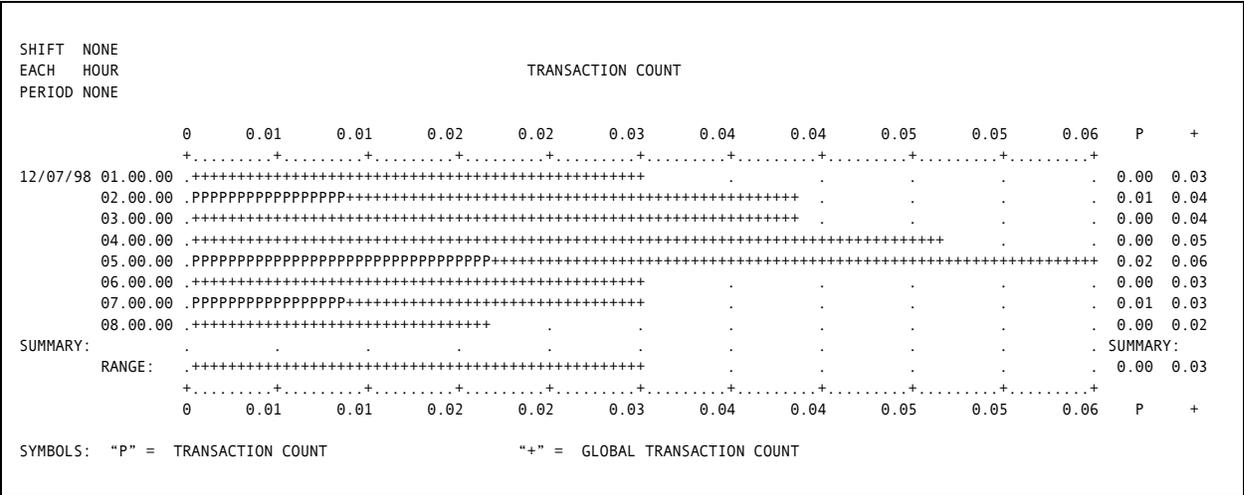
Converting Counts to Rates

Use the RATE option to plot a rate (number per second) rather than a count.

The following commands plot the usage rate for transaction PSPI against the total transaction rate:

```
PLOT TRAN USE RATE TRANID(PSPI) CHAR1(P) CHAR2(+)
FROM 12/07/98 01:00:00
TO 12/07/98 09:00:00
EACH HOUR
END
RUN
```

The preceding commands generate a report like the following:



PLOT2 Command

Use the PLOT2 command together with the PLOT command to plot the performance of two variables side-by-side to see if a cause-and-effect relationship exists between them.

You cannot use the PLOT2 command without the PLOT command.

Option Syntax and Operands

The syntax of the PLOT2 command is as follows:

```
PLOT2 variable [options] [ALONE] [SCALE({n}, {mX}, {COMMON})] [CHAR1(x)]  
[CHAR2(x)]
```

The operands for the PLOT2 command are the same as for the PLOT command except for the COMMON option of the SCALE operand. Specify SCALE(COMMON) if you want the same scale to be used for both the PLOT and PLOT2 variables.

For an explanation of the meaning of the other PLOT2 operands, refer to the explanations of the PLOT operands later in this chapter.

Creating a Simple PLOT2 Report

The following commands generate a report that compares the response time of 0A terminals with the maximum terminal response time for 0A terminals. The average response time and average maximum response time for all terminals will also be plotted.

```
PLOT TERM RESP TERMID(0A++)
PLOT2 TERM MAXRESP TERMID(0A++)
FROM 12/07/98 01:00:00
TO 12/07/98 09:00:00
EACH HOUR
END
RUN
```

The preceding commands generate a report like the following:

SHIFT	NONE	TERMINAL MAXIMUM RESPONSE TIME													
EACH	HOUR	0	0.033	0.066	0.100	0.133	0.166	0.199	0.232	0.266	0.299	0.332	X	-	
PERIOD	NONE	0	0.792	1.584	2.377	3.169	3.961	4.753	5.545	6.338	7.130	7.922	*	+	
12/07/98	01.00.00	.XXXXXXXXXXXX	-----											0.040	0.206
		.*****	-----											0.161	1.599
	02.00.00	.XXXXXXXXXXXX	-----											0.090	0.106
		.*****	-----											2.771	2.771
	03.00.00	.XXXXXXXXXXXX	-----											0.125	0.128
		.*****	-----											1.298	1.298
	04.00.00	.XXXXXXXXXXXX	-----											0.083	0.120
		.*****	-----											1.673	1.673
	05.00.00	.XXXXXXXXXXXX	-----											0.104	0.120
		.*****	-----											4.094	4.094
	06.00.00	.XXXXXXXXXXXX	-----											0.332	0.332
		.*****	-----											6.137	6.137
	07.00.00	.XXXXXXXXXXXX	-----											0.194	0.242
		.*****	-----											6.477	6.477
	08.00.00	.XXXXXXXXXXXX	-----											0.240	0.240
		.*****	-----											7.922	7.922
SUMMARY:													SUMMARY:		
	RANGE:	.XXXXXXXXXXXX	-----											0.151	0.186
		.*****	-----											3.816	3.996
		.*****	-----											0.151	0.186
		.*****	-----											3.816	3.996
		0	0.792	1.584	2.377	3.169	3.961	4.753	5.545	6.338	7.130	7.922	*	+	
		0	0.033	0.066	0.100	0.133	0.166	0.199	0.232	0.266	0.299	0.332	X	-	
SYMBOLS:		"X" = TERMINAL AVERAGE RESPONSE TIME					"- " = GLOBAL TERMINAL AVERAGE RESPONSE TIME								
		"*" = TERMINAL MAXIMUM RESPONSE TIME					"+ " = GLOBAL TERMINAL MAXIMUM RESPONSE TIME								

In the 3600 seconds between 2 a.m. and 3 a.m., PSPI was executed 29 times, which is a rate of 0.01. Globally, 149 transactions were executed in that period, for a rate of 0.04.

Note: A usage rate of zero does not necessarily imply zero usage of the resource. For example, transaction PSPI was used 9 times in the 3600 seconds between 3:00 and 4:00 a.m. However, this yields a usage rate so low ($9 / 3600 = 0.0025$) that it appears as zero in the plot of usage rate.

Vertical Reports (VPLOT, VPLOT2, and MPLOT)

This section discusses operands and sample commands, and provides sample output for the vertical reports generated by the VPLOT, VPLOT 2, and MPLOT commands.

VPLOT Command

Use the VPLOT command to produce reports that plot resource usage in vertical columns. The horizontal axis of the plot represents time.

Option Syntax and Operands

The syntax of the VPLOT command is as follows:

```
VPLOT variable [options] [SCALE({n},{mX})] [STACK=id] [CHAR1(x)] [HDR1(text)] [HDR2(text)] [HDR(text)]
```

The operands for the VPLOT command have the following meanings:

Operand	Meaning
<i>variable</i>	Any report writer variable.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.
SCALE(<i>n</i>) or SCALE(<i>mX</i>) or SCALE(<i>n,mX</i>)	By default, the highest value of a plot's vertical axis equals the highest value reached by a variable in the plot or, for variables representing percentages, 100%. To override this dynamic scale, specify <i>n</i> as the maximum value. If the variable that you choose produces integers, <i>n</i> must be an integer. You can also widen each plot column by using the multiplier <i>mX</i> , where <i>m</i> is an integer.
STACK= <i>id</i>	Replaces the plot character with the identifiers being plotted, from most active to least active starting at the bottom of the plot. You can specify the <i>mX</i> option of the SCALE operand to print more characters of each identifier. For an example using the STACK operand, see later in this chapter.
CHAR1(<i>x</i>)	Sets the plot character. Replace <i>x</i> with any valid EBCDIC character.
HDR1(<i>text</i>)	Overrides the default top heading for the scale of the plot. Replace <i>text</i> with the new heading. To remove the header, replace <i>text</i> with a null string, as in HDR1().
HDR2(<i>text</i>)	Overrides the default bottom heading for the scale of the plot. Replace <i>text</i> with the new column heading. To remove the header, replace <i>text</i> with a null string, as in HDR2().
HDR(<i>text</i>)	Overrides the default description beside the SYMBOLS: heading that describes the plotted variables. Replace <i>text</i> with the new description, not to exceed 30 characters.

Creating a Simple VPLOT Report

The following commands plot terminal response time between 1:00 and 6:00 a.m.:

```
VPLOT TRAN USE
  FROM 12/07/98 01:00:00
  TO   12/07/98 06:00:00
EACH 10 MINUTES
END
RUN
```

The preceding commands generate a report like the following:

SHIFT	NONE	TRANSACTION COUNT			TRAN USE
EACH	10 MINUTES				
PERIOD	NONE				
TRAN USE					TRAN USE
100	I				I 100
	I				I
	I				I
	I				I
	I				I
88	I				I 88
	I				I
	I				I
	I				I
	I				I
75	I				I 75
	I				I
	I			X	I
	I			X	I
	I			X	I
	I			X	I
63	I			X	I 63
	I			X	I
	I			X	I
	I			X	I
	I			XX	I
50	I			XXX	I 50
	I			X XXX	I
	I			X XXX	I
	I			X XXX	I
	I			X XXX	I
38	I	X X X X X X		XX XXX	I 38
	I	X X X X X X	X XXX	XXX	I
	I	X X X X X X	X XXX	XXX	I
	I	X X X X X X	X XXX	XXX	I
	I	X X X X X X	X XXX	XXX	I
AI		---XX--X--X--X--X--X--XXX--XXX			I
25	I	XX X X XXX X XXX XXX			I 25
	I	XX X XX XXX XX XXX XXXX			I
	I	X XX XX XXXXXX XX XXX XXXX			I
	I	X XXXXXX XXXXXX XXXXXX XXXX			I
	I	X XXXXXX XXXXXXXXXXXXXXXXXXXX			I
13	I	X XXXXXX XXXXXXXXXXXXXXXXXXXX			I 13
	I	X XXXXXX XXXXXXXXXXXXXXXXXXXX			I
	I	X XXXXXX XXXXXXXXXXXXXXXXXXXX			I
	I	XXXXXXXXXXXXXXXXXXXXXXXXXXXX			I
	I	XXXXXXXXXXXXXXXXXXXXXXXXXXXX			I
0	II	-----I-----I-----I-----I-----I-----I-----I-----I-----II			I 0
		01:00 02:40 04:20			
		12/07 12/07 12/07			

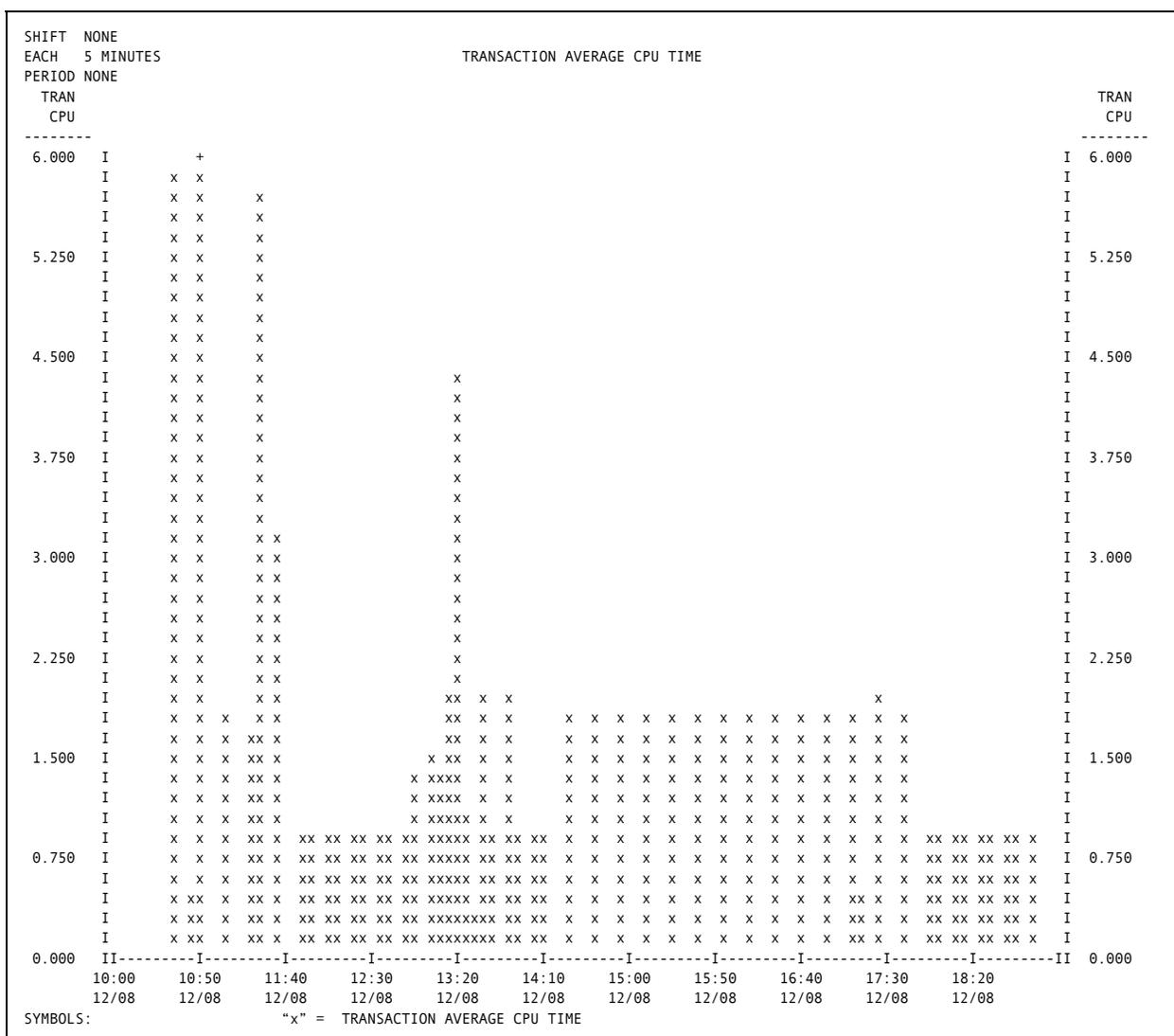
SYMBOLS: "X" = TRANSACTION COUNT

Using the SCALE(n) Operand

The following example uses the SCALE(6.000) operand to plot transaction CPU use with a maximum scale value of 6.000 seconds. The CHAR1 operand specifies x as the plot character.

```
VPLLOT TRAN CPU TOTAL CHAR1(x) SCALE(6.000)
FROM 12/08/98 10:00:00
TO 12/08/98 19:00:00
EACH 5 MINUTES
END
RUN
```

The preceding commands generate a report like the following:



VPLOT2 Command

Use the VPLOT2 command with the VPLOT command to plot a second variable directly over the first one, allowing you to compare the performance of two variables to see if a cause-and-effect relationship exists between them.

You cannot use the VPLOT2 command without the VPLOT command.

Option Syntax and Operands

The syntax of the VPLOT2 command is as follows:

```
VPLOT2 variable [options] [SCALE({n},{mk},{COMMON})] [CHAR1(x)] [HDR1(text)] [HDR2(text)] [HDR(text)]
```

The operands for the VPLOT2 command are the same as for the VPLOT command except that the VPLOT2 command does not support the STACK operand and it does support the COMMON option of the SCALE operand. Specify SCALE(COMMON) if you want the same scale to be used to plot both variables.

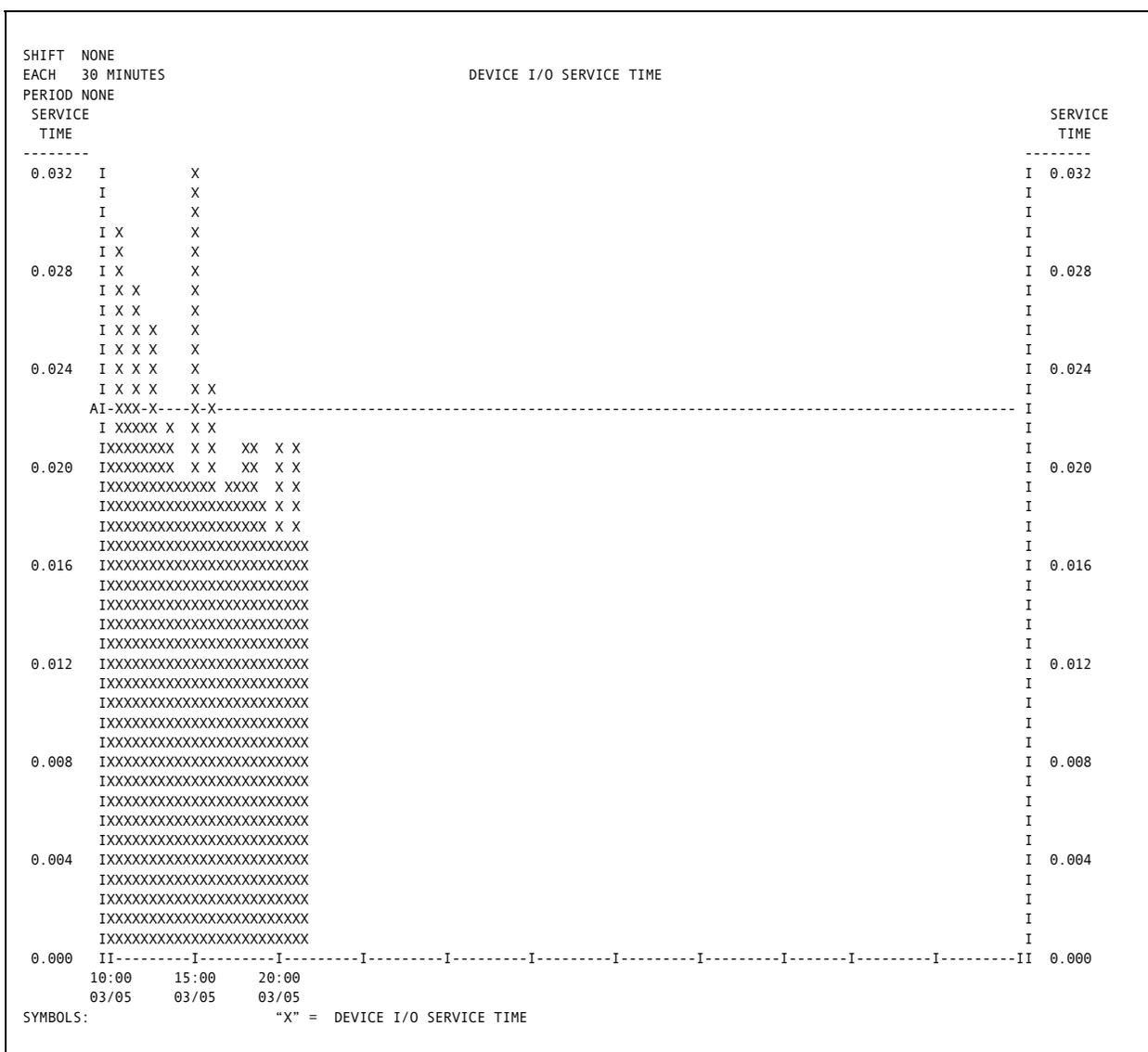
For an explanation of the other VPLOT2 operands, refer to the descriptions of the VPLOT operands listed earlier in this chapter.

Creating a Simple VLOT2 Report

The following example uses the VLOT and VLOT2 commands to plot both transaction use and terminal response time:

```
VLOT TRAN USE SCALE(50) CHAR1(X)
VLOT2 TERM RESP SCALE(1.000) CHAR1(O)
EACH 5 MINUTES
FROM 12/07/98 00:00:00
TO 12/07/98 09:00:00
END
RUN
```

The preceding commands generate a report like the following:



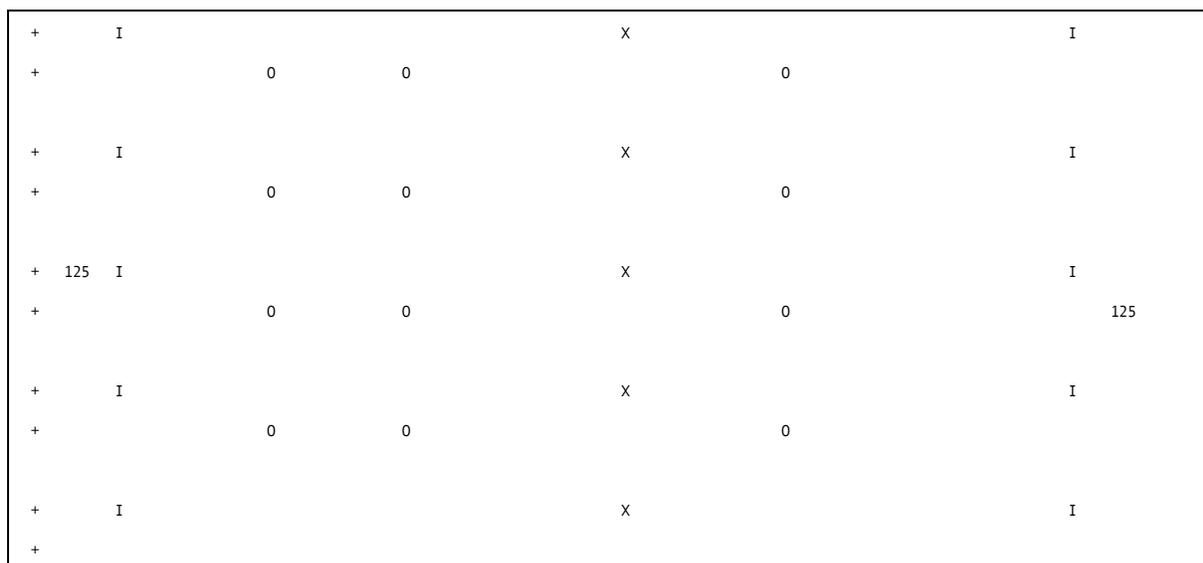
Using the SCALE(COMMON) Operand

The following commands plot transaction use and dataset use. The SCALE(COMMON) operand causes the same scale to be used for both variables being plotted.

```
VPLLOT TRAN USE CHAR1(X)
VPLLOT2 DATASET USE CHAR1(O) SCALE(COMMON)
FROM 12/07/98 00:00:00
TO 12/07/98 09:00:00
EACH 5 MINUTES
END
RUN
```

The preceding commands generate a report like the following, shown only in part.

SHIFT NONE				TRANSACTION COUNT			
EACH 5 MINUTES							
PERIOD NONE							
TRAN	USE					FILE	USE
-----						-----	
+	166 I			X		I	
+		+	+				166
+	I			X		I	
+		0	0				
+	I			X		I	
+		0	0				
+	I			X		I	
+		0	0				
+	I			X		I	
+		0	0				
+	146 I			X		I	
+		0	0				146
+	I			X		I	
+		0	0		0		
+	I			X		I	
+		0	0		0		



Distribution Plot Reports (MPLOT)

This section discusses operands and sample commands, and provides sample output for the distribution reports generated by the MPLOT command.

MPLOT Command

Use the MPLOT command to produce a vertical plot similar to that produced by the VPLOT command, except that only the outline of the columns is printed.

MPLOT allows you to request multiple variables on the same report.

Option Syntax and Operands

The syntax of the MPLOT command is as follows:

```
MPLOT variable [options] [SCALE({n},{mX})] [CHAR1(x)] [HDR1(text)] [HDR2(text)] [HDR(text)] [AND,
variable [options] [SCALE({n},{mX})] [CHAR1(x)] [HDR1(text)] [HDR2(text)] [HDR(text)] [AND,...]
```

The operands for the MPLOT command have the following meanings:

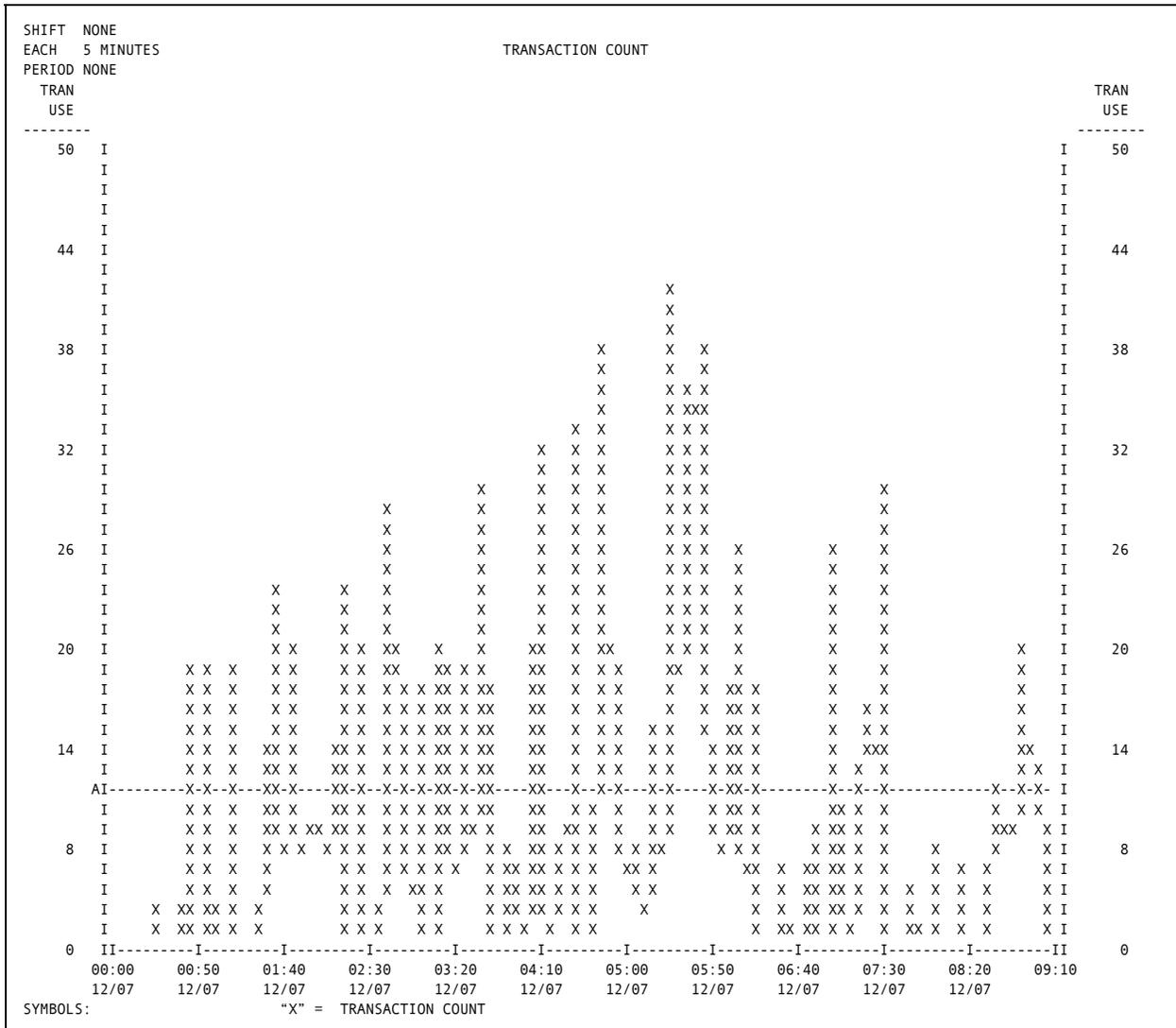
Operand	Meaning
<i>variable</i>	Any report writer variable.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.
SCALE(<i>n</i>) or SCALE(<i>mX</i>) or SCALE(<i>n,mX</i>)	By default, the highest value of a plot's horizontal axis is set to the highest value of the plotted variables. For variables that represent percentages, the default maximum value is 100%. To override this dynamic scale, specify <i>n</i> as the maximum value. If the variable that you choose produces integers, <i>n</i> must be an integer. You cannot use more than three digits to the right of the decimal point when specifying <i>n</i> . You can also widen the outline of the plot by using the multiplier <i>mX</i> , where <i>m</i> is an integer.
CHAR1(<i>x</i>)	Sets the plot character. Replace <i>x</i> with any valid EBCDIC character.
HDR1(<i>text</i>)	Overrides the default top heading for the scale of the plot. Replace <i>text</i> with the new heading. To remove the header, replace <i>text</i> with a null string, as in HDR1().
HDR2(<i>text</i>)	Overrides the default bottom heading for the scale of the plot. Replace <i>text</i> with the new column heading. To remove the header, replace <i>text</i> with a null string, as in HDR2().
HDR(<i>text</i>)	Overrides the default description beside the SYMBOLS: heading that describes the plotted variables. Replace <i>text</i> with the new description, not to exceed 30 characters.
AND,	Signals that another variable will be specified. The variable will be plotted on top of the preceding variables. The comma after <i>AND</i> lets you continue to the next line.

Using One Variable

The following example uses the MPLLOT command to plot terminal response times:

```
MPLLOT TRAN USE SCALE(50)
EACH 5 MINUTES
FROM 12/07/98 00:00:00
TO 12/07/98 23:59:59
END
RUN
```

The preceding commands generate a report like the following:



Using Multiple Variables

The following example uses the MPLOT command to plot transaction use with Xs and terminal response time with Os:

```
MPLOT TRAN USE SCALE(50) CHAR1(X) AND,  
      TERM RESP SCALE(1.000) CHAR1(O)  
      EACH 5 MINUTES  
      FROM 12/07/98 00:00:00  
      TO 12/07/98 23:59:59  
END  
RUN
```

The preceding commands generate a report like the following, shown only in part.

The following commands are the same as those in the preceding example except for the addition of a third variable, DATASET USE, which will be plotted with plus signs (+):

```
FROM 12/07/98 00:00:00
TO   12/07/98 09:00:00
EACH 5 MINUTES
MPLOT TRAN USE    SCALE(50)   CHAR1(X) AND,
      TERM RESP  SCALE(1.000) CHAR1(O) AND,
      DATASET USE SCALE(75)   CHAR1(+)
```

END
RUN

The preceding commands generate a report like the following, shown only in part.

Horizontal Reports (HPLOT)

The following section discusses operands and sample commands and provides sample output for HPLOT reports.

HPLOT Command

Use the HPLOT command to plot the distribution of one variable based on the value of another variable. The variable you specify with the HPLOT command is plotted along the horizontal axis. The second variable, specified with either the VPLOT or the MPLOT command, is plotted vertically.

You cannot use the HPLOT command without either the MPLOT or VPLOT command. The HPLOT command must follow the MPLOT or VPLOT command.

To set the plot character for an HPLOT report, specify the CHAR1 operand with the MPLOT or VPLOT command.

Option Syntax and Operands

The syntax of the HPLOT command is as follows:

```
HPLOT variable [options] [SCALE(n)] [SCATTER]
```

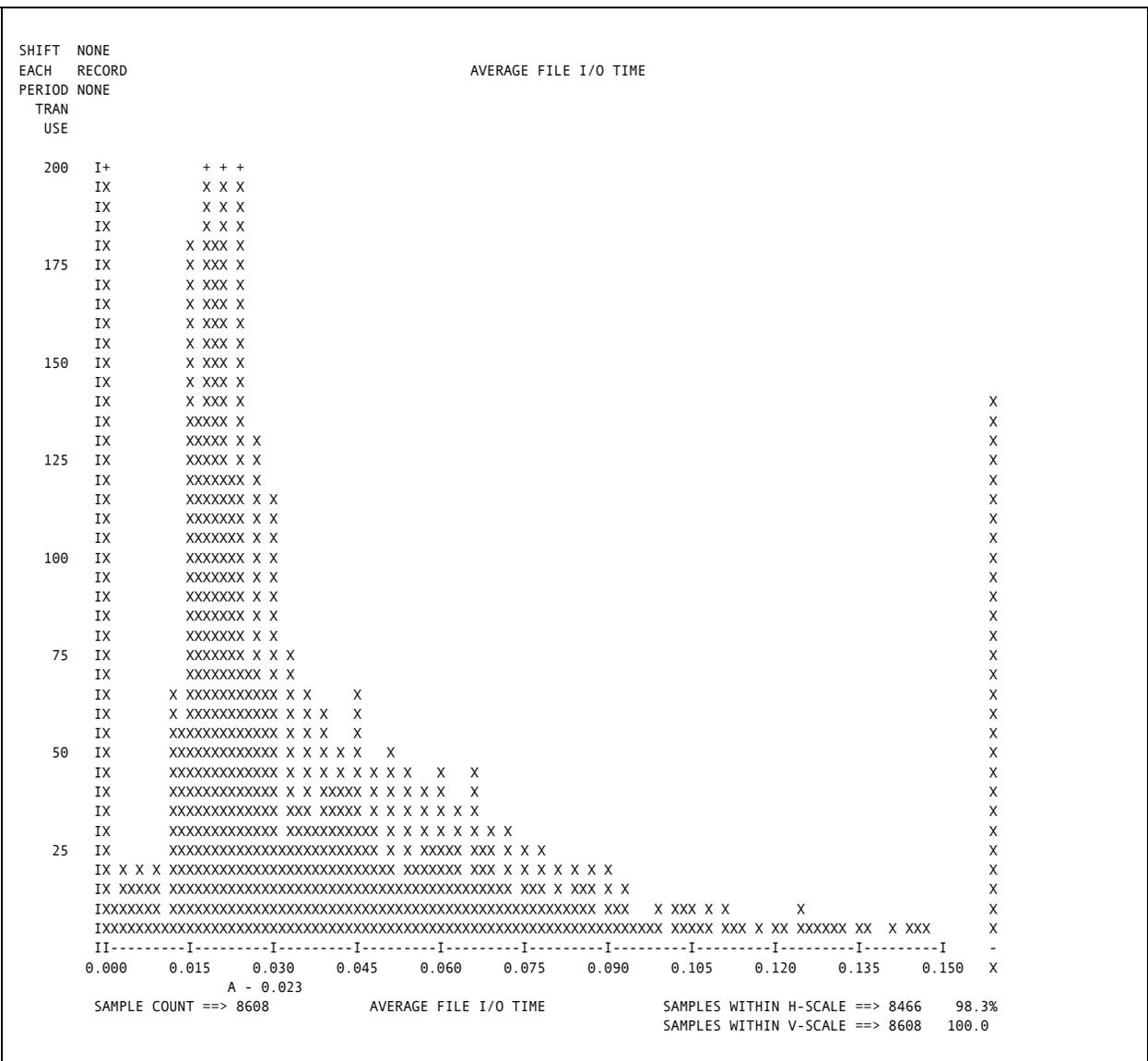
The operands for the HPLOT command have the following meanings:

Operand	Meaning
<i>variable</i>	Any report writer variable.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.
SCALE(<i>n</i>)	By default, the highest value of a plot's horizontal axis is set to the highest value of the plotted variables. For variables that represent percentages, the default maximum value is 100%. To override this dynamic scale, specify <i>n</i> as the maximum value. If the variable that you choose produces integers, <i>n</i> must be an integer. You cannot use more than three digits to the right of the decimal point when specifying <i>n</i> . If you use the RECORD option of the EACH command, you must use SCALE(<i>n</i>), specifying a value for <i>n</i> appropriate for the activity you are plotting.
SCATTER	Prints a scatter plot showing how many times each point in the plot is reached.

The following commands are similar to those in the previous example but the interval has been changed from each five minutes to each record and the SCALE operand has been specified with both the V PLOT and H PLOT commands. Whenever the command EACH RECORD is used, the SCALE operand must be specified with the H PLOT command.

```
V PLOT TRAN USE      SCALE(200)  CHAR1(X)
H PLOT FILE SERVTIME SCALE(0.150)
  EACH RECORD
END
RUN
```

The preceding commands generate a report like the following:



Graph Reports

Graph reports are graphs used to compare related identifiers. This chapter describes how to create graph reports using the GRAF and GRAF2 commands, and provides examples.

For more information on related topics, refer to the following chapters:

- See the chapter titled “Commands” to become familiar with report writer commands.
- For an explanation of the structure of graph reports, see the chapter titled “Reading Reports.”
- For descriptions of report writer variables and lists of the ID options you can use with each class of variable, see the chapter titled “Variables.”
- For descriptions of ID options and their use with commands and variables, see the chapter titled “ID Options.”

GRAF Reports (GRAF and GRAF2)

Syntax and operands used to create GRAF reports are described in the next section. Sample output is also included.

GRAF Command

The GRAF command produces a single-line bar graph of the performance of each identifier associated with a variable. For example, using the GRAF command with a TRANSACTION variable produces a bar graph of the variable for each transaction.

Command Syntax and Operands

The syntax of the GRAF command is as follows:

```
GRAF [idoption[(identifiers)]] variable [options] [SCALE({n},{mX)}] [SORT order] [FOR n] CHAR1(x)
```

The following list describes the operands used with the GRAF command:

Operand	Meaning
<i>idoption</i> (<i>identifiers</i>)	<p>Any ID option that is valid for the variable, specified with or without identifiers. Specifying this operand causes the resulting report to be organized by the specified ID option and limited to the identifiers specified by <i>identifiers</i>. If you do not specify an ID option, all identifiers of the type associated with the variable are included in the report.</p> <p>For example, a graph of a TRANsaction variable by default graphs the value of that variable for each transaction. Specifying the ID option and identifier TRANID(TRAN1) graphs the value of the variable for transaction TRAN1 only; specifying the ID option TERMID graphs the value for each terminal.</p>
<i>variable</i>	Any report writer variable.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.

Operand	Meaning
SCALE(<i>n</i>) or SCALE(<i>mX</i>) or SCALE(<i>n,mX</i>)	<p>By default, the highest value of a bar graph's axis is set to the highest value of the graphed variables. For variables that represent percentages, the default maximum value is 100%.</p> <p>To override this dynamic scale, specify <i>n</i> as the maximum value. If the variable that you choose produces integers, <i>n</i> must be an integer. You cannot use more than three digits to the right of the decimal point when specifying <i>n</i>.</p> <p>In some cases, bar graphs in a graph report will be too short to appear. To display these graphs, you can cause each plot character to appear as <i>m</i> characters by specifying the multiplier <i>mX</i>, where <i>m</i> is an integer. Since the resulting bar graphs will no longer correspond numerically to the scale, this is useful only for determining relative performance.</p>
<i>order</i>	<p>Sorts the report. Replace <i>order</i> with one of the following:</p> <p>DESCENDING – Sorts identifiers in descending order of value of the variable specified. This is the default sorting order.</p> <p>ASCENDING – Sorts identifiers in ascending order of value of the variable specified.</p>
FOR <i>n</i>	<p>Limits the report to the <i>n</i> identifiers in each time that would appear first in the report. For reports sorted in descending order, these are the <i>n</i> most active identifiers; for reports sorted in ascending order, these are the <i>n</i> least active identifiers. The value for <i>n</i> must be an integer.</p>
CHAR1(<i>x</i>)	<p>Sets the plot character. Replace <i>x</i> with any valid EBCDIC character.</p>

Changing the Scale of a Report

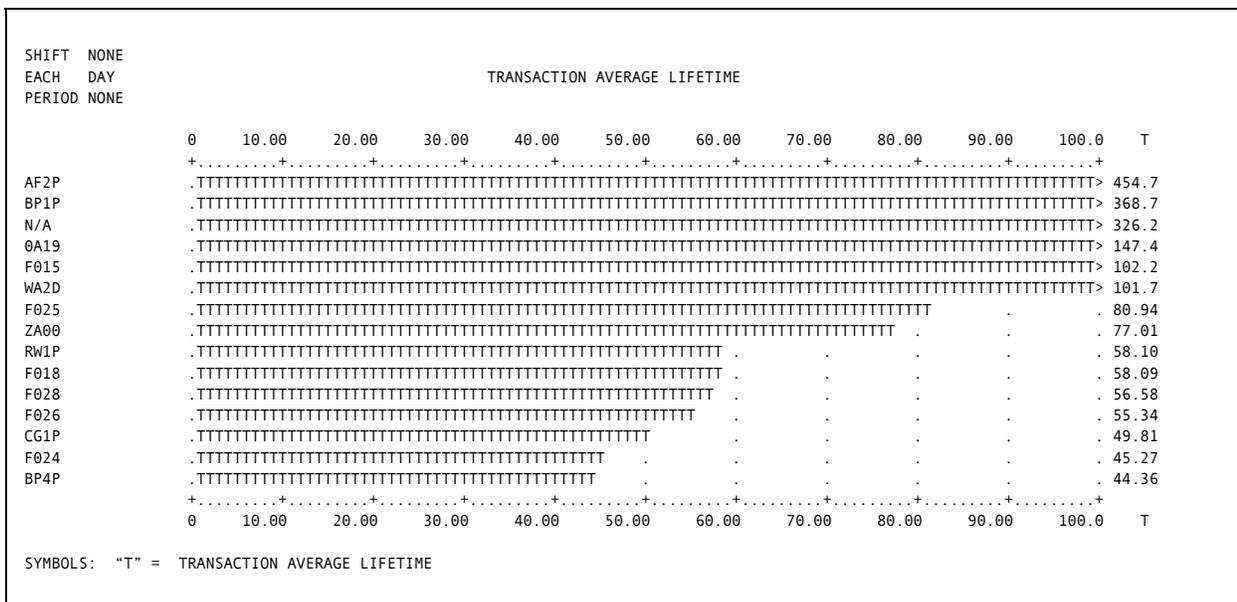
Use the SCALE(*n*) operand to alter the scale of a graph report.

If the variable does not yield results with a decimal point, you cannot use a decimal point in the SCALE operand.

The following commands produce a report similar to the preceding one but with a scale of 100.0. The FOR operand limits the report to 15 terminals.

```
GRAF TERMID TRAN LIFE FOR 15 CHAR1(T) SCALE(100.0)
END
RUN
```

The preceding commands generate a report like the following:



When the value of the variable being plotted exceeds the scale used, a greater-than sign (>) appears to the right of the graph for the resource, as is the case for the first several transactions in the preceding report.

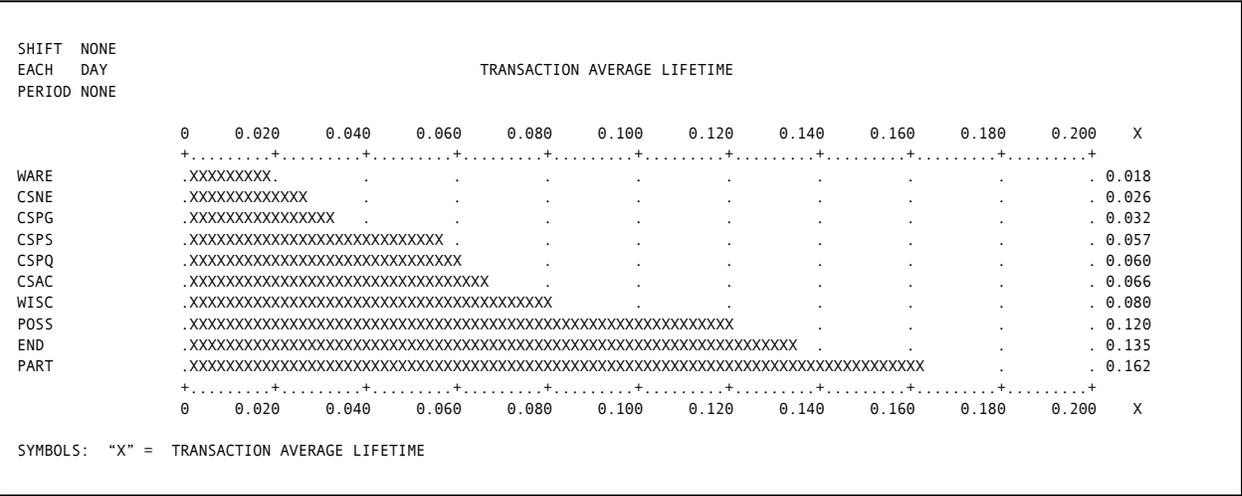
Sorting Reports

Use the SORT ASCENDING operand to sort a graph report in ascending order.

The following commands produce a report on the ten transactions with the shortest average transaction lifetimes:

```
GRAF TRAN LIFETIME FOR 10 SORT ASCENDING SCALE(0.200)
END
RUN
```

The preceding commands generate a report like the following:



Creating Reports on Specified Shifts

Use the SHIFT command to create reports that cover only certain hours of the day.

The following commands produce a report on the five transactions active between 11:00 p.m. and 7:00 a.m. during the week of January 5, 2000, that have names starting with the letter C, and have the shortest lifetimes:

```
GRAF TRAN LIFETIME SCALE(1.0) TRANID(C*),
      FOR 5 SORT ASCENDING
      FROM 01/05/00 00.00.00
      TO   01/11/00 00.00.00
      SHIFT 23:00:00 07:00:00
      END
      RUN
```

The preceding commands generate a report like the following partial sample:

	0	0.100	0.200	0.300	0.400	0.500	0.600	0.700	0.800	0.900	1.000	X
CSTA	+	+	+	+	+	+	+	+	+	+	+	+
CSSN	.XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX 0.354
	+	+	+	+	+	+	+	+	+	+	+	+
	0	0.100	0.200	0.300	0.400	0.500	0.600	0.700	0.800	0.900	1.000	X

SYMBOLS: "X" = TRANSACTION LIFETIME IN SECONDS.

Although FOR 5 is used in the above report, only two transactions appear. Those transactions are the only ones that were used during the 11:00 p.m. to 7:00 a.m. shift whose names started with the letter C.

Limiting Input to a Specified Range

Use the RANGE command to restrict a report to resources whose activity falls within an average lifetime. The RANGE command is explained in further detail in the chapter titled "Commands."

The following commands produce a graph of the lifetime of transactions with an average lifetime of 1.000 second or less. Because the DEFERRED operand is specified with the RANGE command, the range criterion is not checked until all data has been read, enabling the report writer to determine the average lifetime of each transaction.

```
GRAF TRAN LIFETIME SCALE(1.000)
  FROM 12/08/00 10:00:00
  TO   12/08/00 18:00:00
  RANGE TRAN LIFE 0.000-1.000 DEFERRED
END
RUN
```

The preceding commands produce a report like the following:

SHIFT	NONE	TRANSACTION AVERAGE LIFETIME											
EACH	DAY	0	0.100	0.200	0.300	0.400	0.500	0.600	0.700	0.800	0.900	1.000	X
PERIOD	NONE	+.....+.....+.....+.....+.....+.....+.....+.....+.....+.....+.....+											
CSSF		.XX 0.767
END		.XX 0.403
VGBR		.XX 0.371
CSNE		.XX 0.211
PSPI		.XX 0.149
POSS		.XXXXXXXXXX. 0.092
CSAC		.XXXX 0.038
CSPQ		.XXX 0.032
CSGM		.XXX 0.028
		+.....+.....+.....+.....+.....+.....+.....+.....+.....+.....+.....+											
		0	0.100	0.200	0.300	0.400	0.500	0.600	0.700	0.800	0.900	1.000	X
SYM													

GRAF2 Reports

Use the GRAF2 command to produce one or more sub-graphs for each individual resource included by the GRAF command. These sub-graphs can do either of the following:

- Graph the performance of a second, related set of resources. For example, if your GRAF command graphs by terminal, a GRAF2 command could list each transaction used by each terminal and show the average terminal response time for each.
- Graph the values of one or more variables for each resource. For example, if your GRAF command graphs by transaction, a GRAF2 command could show the usage, CPU use, and average lifetime of each transaction.

GRAF2 Command

You cannot use the GRAF2 command without the GRAF command.

The variables you specify with the GRAF and GRAF2 commands must be logically related. The following table shows incorrect and correct uses of the GRAF and GRAF2 commands:

Incorrect:	Correct:
GRAF DATASET REQUESTS GRAF2 TRANSACTION USE	GRAF TRANSACTION USE GRAF2 TERMINAL IO
GRAF DATASET SERVTIME GRAF2 PROGRAM USE	GRAF TERMINAL RESP GRAF2 TRANSACTION USE
GRAF TERMINAL IO GRAF2 VARS TERM USE AND, PROGRAM USE AND, DATASET IO AND, TRANSIENT USE	GRAF TRANSACTION USE GRAF2 VARS TERM USE AND, PROGRAM USE AND, DATASET IO AND, TRANSIENT USE
GRAF TERMINAL IO GRAF2 VARS TERM USE AND, TRANSACTION USE	GRAF TERMINAL IO GRAF2 VARS TERM USE AND, TERM RESP AND, TERM MAXRESP AND, TERM BYTESREAD

Command Syntax and Operands

The syntax of the GRAF2 command is as follows:

```
GRAF2 [VARS] [idoption(identifiers)] variable [options] [HDR(text)] [CHAR1(x)] [SCALE({n},{mK},{COMMON})] [SORT order]
[FOR n] [AND,
      [variable [options] [HDR(text)] [CHAR1(x)] [AND,...]
```

The operands for the GRAF2 command have the following meanings:

Operand	Meaning
VARS	Indicates that there is more than one variable in this GRAF2 report.
<i>idoption</i> (<i>identifiers</i>)	Any ID option that is valid for the variable, specified with or without identifiers. Specifying this operand causes the resulting sub-graphs to be organized by the specified ID option and limited to the identifiers specified by <i>identifiers</i> . If you do not specify an ID option, all identifiers of the type associated with the variable are included in the sub-graphs. You can use this operand only if you specify a single variable with the GRAF2 command.
<i>variable</i>	Any report writer variable.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.
HDR(<i>text</i>)	Overrides the default heading for the variable. Replace <i>text</i> with a heading of 15 or fewer characters.
CHAR1(<i>x</i>)	Sets the plot character for the variable. Replace <i>x</i> with any valid EBCDIC character.

Operand	Meaning
SCALE(<i>n</i>) or SCALE(<i>mX</i>) or SCALE(<i>n,mX</i>) or SCALE(COMMON)	<p>By default, the highest value of a bar graph's axis is set to the highest value of the graphed variables. For variables that represent percentages, the default maximum value is 100%.</p> <p>To override this dynamic scale, specify <i>n</i> as the maximum value. If the variable that you choose produces integers, <i>n</i> must be an integer. You cannot use more than three digits to the right of the decimal point when specifying <i>n</i>.</p> <p>In some cases, bar graphs in a graph report will be too short to appear. To display these graphs, you can cause each plot character to appear as <i>m</i> characters by specifying the multiplier <i>mX</i>, where <i>m</i> is an integer. Since the resulting bar graphs will no longer correspond numerically to the scale, this is useful only for determining relative performance.</p> <p>To have the same scale used for both variables, specify SCALE(COMMON).</p> <p>You can specify the SCALE operand only with the first variable you specify with the GRAF2 command.</p>
<i>order</i>	<p>Sorts the report. Replace <i>order</i> with one of the following:</p> <p>DESCENDING – Sorts identifiers in descending order of value of the variable specified. This is the default sorting order.</p> <p>ASCENDING – Sorts identifiers in ascending order of value of the variable specified.</p> <p>You can specify the SORT operand only with the first variable you specify with the GRAF2 command.</p>
FOR <i>n</i>	<p>Limits the report to the <i>n</i> identifiers in each time that would appear first in the report, where <i>n</i> is an integer. For reports sorted in descending order, these are the <i>n</i> most active identifiers; for reports sorted in ascending order, these are the <i>n</i> least active identifiers. You can specify the FOR operand only with the first variable you specify with the GRAF2 command.</p>
AND,	<p>Signals that another variable will be specified. The comma after AND lets you continue to the next line. If you specify multiple variables, you must use the VARS operand.</p>

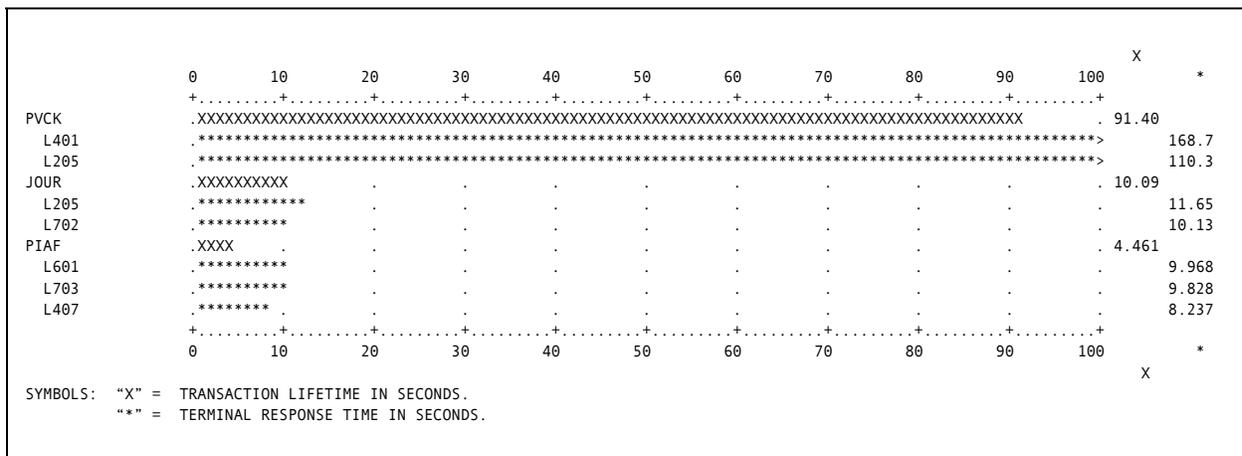
Tailoring GRAF2 Reports Using the SCALE Operand

Use the SCALE operand to specify a scale for the GRAF2 variable. This operand is particularly useful for creating a common scale in reports on time-related variables such as TRANsaction LIFETIME and TERMinal RESPonsetime.

The following commands are the same as those in the previous example except that the SCALE operand is included to produce a report that uses the same scale for both the GRAF and GRAF2 variables, and the default plot characters are used:

```
GRAF TRANSACTION LIFE FOR 3 SCALE(100.0)
GRAF2 TERMINAL RESP FOR 2 SCALE(100.0)
END
RUN
```

The preceding commands generate a report like the following partial sample:



Comparing Performance Using Multiple-Variable Criteria

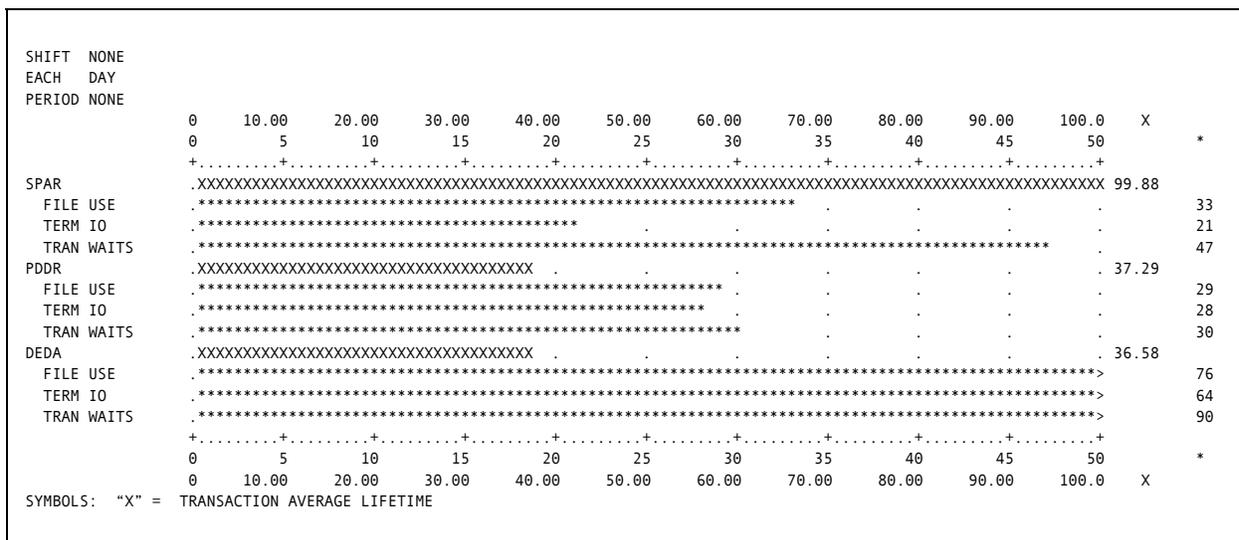
Use multiple variables with the GRAF2 command to compare the performance of a class of resources with respect to those variables. For example, you can compare the performance of transactions with respect to terminal use, terminal response time, and dataset I/O.

When you specify more than one variable with the GRAF2 command, a one-line bar graph is created for each variable specified. The bar graph represents *average* performance for all active identifiers associated with the variable.

The following commands produce a report on the three transactions with the longest lifetimes. For each transaction, the number of file requests, terminal I/Os, and transaction waits per use will also be plotted, using a scale of 50. The VARS operand precedes the variables specified with the GRAF2 command.

```
GRAF      TRAN LIFE      SCALE(100) FOR 3,
GRAF2 VARS DATASET USE  SCALE(50) AND,
          TERM IO        AND,
          TRAN WAITS
END
RUN
```

The preceding commands generate a report like the following partial sample:



Flashback Reports

Flashback reports are tabular reports that contain detailed information about transactions. This chapter describes how to create flashback reports using the FLASHBACK LIST and FLASHBACK DETAIL commands, and provides examples for both.

The FLASHBACK command has two forms:

- FLASHBACK LIST generates one line of output per transaction.
- FLASHBACK DETAIL generates one page of output per transaction.

Transactions are presented in chronological order of ending time. You cannot sort flashback reports.

For more information on related topics, refer to the following chapters:

- See the chapter titled “Commands” to become familiar with report writer commands.
- For an explanation of the structure of flashback reports, see the chapter titled “Reading Reports.”
- For descriptions of report writer variables and lists of the ID options you can use with each class of variable, see the chapter titled “Variables.”
- For descriptions of ID options and their use with commands and variables, see the chapter titled “ID Options.”

WARNING! *Each flashback report must be run as a single report in a separate run. You cannot specify another primary report command (such as TAB, PLOT, or SELECT) in a job stream with a FLASHBACK LIST or FLASHBACK DETAIL command.*

Output Considerations

FLASHBACK commands can generate enormous amounts of output if run against a very large input file. To avoid problems, use either the FROM and TO commands, the SELECT command, or the COUNT command to limit output.

FLASHBACK LIST Reports

The following section discusses operands and sample commands and provides sample output for FLASHBACK LIST report.

FLASHBACK LIST Command

Use the FLASHBACK LIST command to produce a tabular report containing one line of output per transaction.

Command Syntax and Operands

The syntax of the FLASHBACK LIST command is as follows:

```
FLASHBACK LIST [columns] [idoption[(identifiers)] [variable [options] [HDR1(text)] [HDR2(text)] [AND,
variable [options] [HDR1(text)] [HDR2(text)] [AND,...]]]
```

The following list describes the FLASHBACK LIST command operands:

Operand	Meaning
<i>columns</i>	<p>One or more of the following, in any order, to indicate the first columns to appear in the report. If you do not specify any of these columns, the resulting report is organized by region ID and time, as shown in the default FLASHBACK LIST report.</p> <p>DATETIME Creates columns for the date and time of the record.</p> <p>DATE Creates a column for the date of the record.</p> <p>TIME Creates a column for the time of the record.</p> <p>DAY Creates a column for the day of the record (MONDAY, TUESDAY, and so forth).</p> <p>JOBNAME Creates a column for the CICS jobnames.</p> <p>REGID Creates a column for the CICS region IDs. Region IDs are set in the EXPCMIT with the REGID parameter.</p> <p>You can enter only one of DATETIME, DATE, TIME, and DAY. Because the width of the output page limits the number of columns that can appear in a report, you can use one of these operands only if you specify the variables to be included in your report and limit their number. The number of variables you can specify depends upon the width of the column for each.</p>
<i>idoption</i> (<i>identifiers</i>)	Lists all the resources (or just those specified by <i>identifiers</i>) for an ID option. You can specify multiple ID options.

Operand	Meaning
<i>variable</i>	Any report writer variable. If you specify a variable, you must specify at least one ID option.
<i>options</i>	Any valid variable options or ID options. Each ID option you specify must include at least one identifier.
HDR1(<i>text</i>)	Overrides the default top heading for a column. Replace <i>text</i> with the new column heading. To remove the header, replace <i>text</i> with a null string, as in HDR1().
HDR2(<i>text</i>)	Overrides the default bottom heading for a column. Replace <i>text</i> with the new column heading. To remove the header, replace <i>text</i> with a null string, as in HDR1().
AND,	Signals that another variable will be specified. The comma after AND lets you continue to the next line.

Default FLASHBACK LIST Report

To create a default flashback report, specify the command FLASHBACK LIST without any ID options or variables.

You should use either the COUNT command or the FROM and TO commands to limit report output.

The following commands produce the default FLASHBACK LIST report. The FROM and TO commands limit the report to a short time period.

```
FLASHBACK LIST
  FROM 12/06/03 21:22:00
  TO   12/06/03 21:24:00
END
RUN
```

The preceding commands generate a report like the following partial sample:

CICS ID	END TIME	TERM ID	OPER ID	TRAN ID	TRAN NUMBER	TRAN LIFE	TRAN CPU	TERM RESP	TERM IO	TRAN I/O	FILE COUNT	FILE USE	PROG COUNT	ABEND CODE
B1	21.22.41	N/A	N/A	TCP	0	911.9	2.110	0.000	0	0.381	1	241	11	
B1	21.22.42	N/A	N/A	KCP	0	912.3	1.878	0.000	0	0.000	0	0	0	
B1	21.23.07	N/A	N/A	JJJ	0	937.1	1.942	0.000	0	0.000	0	0	0	
B1	21.23.07	N/A	N/A	CATD	51	6.529	0.326	0.000	0	0.000	0	0	5	
B1	21.23.08	N/A	N/A	CSNE	52	0.081	0.068	0.000	0	0.000	0	0	3	
B1	21.23.09	ZA00	N/A	CSGM	53	0.264	0.010	0.013	2	0.000	0	0	1	
B1	21.23.15	N/A	N/A	VGBR	55	0.332	0.023	0.000	0	0.000	0	0	2	
B1	21.23.21	N/A	N/A	CSNE	56	0.013	0.002	0.000	0	0.000	0	0	3	
B1	21.23.41	X250	N/A	VGBQ	57	19.33	0.151	0.389	5	0.000	0	0	7	AKCT
21.23.41					324	309.7	0.723	0.201	7	0.042	0	241	4	

The preceding report contains the following information:

Heading	Meaning
CICS ID	Region ID in which the data were collected
END TIME	Time the transaction completed
TERM ID	ID of the terminal from which the transaction was executed
OPER ID	ID of the operator who executed the transaction
TRAN ID	Transaction ID
TRAN NUMBER	Transaction number
TRAN LIFE	Transaction lifetime (seconds)
TRAN CPU	Transaction CPU time (seconds)
TERM RESP	Terminal average response time (seconds)
TERM IO	Number of terminal I/Os by the transaction
TRAN I/O	Total I/O time for the transaction
FILE COUNT	Number of files accessed by the transaction
FILE USE	Number of file I/Os by the transaction
PROG COUNT	Number of programs used by the transaction
ABEND CODE	Abend code for the transaction, if any

Using ID Options to Limit FLASHBACK LIST Reports

To limit the information contained in a flashback report to specific resources, such as a single operator or those transactions beginning with the letters *ACCT*, specify an ID option and identifiers with the FLASHBACK LIST command.

The following commands produce a FLASHBACK LIST report showing only the transactions of the operator SWS:

```
FLASHBACK LIST OPERID(SWS)
END
RUN
```

The preceding commands generate a report like the following partial sample:

CICS ID	END TIME	TERM ID	OPER ID	TRAN ID	TRAN NUMBER	TRAN LIFE	TRAN CPU	TERM RESP	TERM IO	TRAN I/O	FILE COUNT	FILE USE	PROG COUNT	ABEND CODE
B1	00.45.29	0A02	SWS	POSS	67	1.011	0.274	1.011	2	0.000	0	0	3	
B1	00.45.35	0A02	SWS	POSS	69	2.387	0.051	1.187	3	0.199	1	2	6	
B1	00.45.38	0A02	SWS	POSS	70	0.419	0.249	0.000	1	0.000	0	0	3	
B1	00.45.44	0A02	SWS	DEDA	71	5.040	0.705	0.992	4	0.034	1	1	4	
B1	00.45.44	0A02	SWS	POSS	72	0.078	0.023	0.034	2	0.039	1	1	4	
B1	00.46.03	0A02	SWS	POSS	73	0.025	0.002	0.025	2	0.000	0	0	3	
B1	00.46.12	0A02	SWS	SPAM	75	1.422	0.291	0.462	3	0.040	1	1	4	
B1	00.46.18	0A02	SWS	SPAM	76	0.446	0.085	0.199	3	0.052	1	1	4	
B1	00.46.37	0A02	SWS	SPAM	77	0.094	0.020	0.000	1	0.048	1	1	4	
B1	00.46.38	0A02	SWS	POSS	78	0.088	0.025	0.033	2	0.038	1	1	4	
B1	00.46.40	0A02	SWS	POSS	79	0.034	0.021	0.000	1	0.000	0	0	3	
B1	00.58.19	0A02	SWS	PBLM	80	699.2	1.925	0.109	156	0.030	1	1	4	
B1	00.58.19	0A02	SWS	POSS	81	0.091	0.026	0.041	2	0.047	1	1	4	
B1	05.28.09	0A02	SWS	POSS	574	0.028	0.017	0.000	1	0.000	0	0	3	
B1	05.28.09	0A02	SWS	PSPI	575	0.047	0.016	0.009	2	0.000	0	0	3	
B1	05.28.19	0A02	SWS	PSPI	577	0.177	0.054	0.044	4	0.000	0	0	3	
B1	05.28.29	0A02	SWS	PSPI	580	0.170	0.049	0.029	5	0.000	0	0	3	
B1	05.28.41	0A02	SWS	PSPI	581	0.152	0.050	0.024	5	0.000	0	0	3	
B1	05.28.53	0A02	SWS	PSPI	582	0.148	0.049	0.024	5	0.000	0	0	3	
B1	05.29.03	0A02	SWS	PSPI	584	0.154	0.049	0.025	5	0.000	0	0	3	
B1	05.29.20	0A02	SWS	PSPI	588	0.118	0.040	0.018	5	0.000	0	0	3	
B1	05.29.37	0A02	SWS	PSPI	592	0.100	0.045	0.013	5	0.000	0	0	3	
B1	05.29.48	0A02	SWS	PSPI	593	0.130	0.046	0.018	5	0.000	0	0	3	
B1	05.32.32	0A02	SWS	PSPI	603	0.219	0.051	0.039	5	0.000	0	0	3	
B1	05.34.14	0A02	SWS	PSPI	606	0.129	0.032	0.129	2	0.000	0	0	3	
05.34.14					8003	28.47	0.168	0.132	231	0.021	0	10	3	

The following commands produce a transaction log:

```
FLASHBACK LIST REGID JOBNAME DATETIME TRANID,
                OPERID,
                TERMID,
                TRAN NUM
```

```
END
RUN
```

The preceding commands generate a report like the following partial sample:

ID	JOBNAME	DATE	TIME	TRAN ID	OPER ID	TERM ID	TRAN NUMBER
A1	CICA	12/06/94	21.16.58	CSDA	N/A	CNSL	19
A1	CICA	12/06/94	21.21.52	CIRB	N/A	CNSL	20
B1	CICB	12/06/94	21.22.41	TCP	N/A	N/A	0
A1	CICA	12/06/94	21.24.20	TCP	N/A	N/A	0
B1	CICB	12/06/94	21.22.42	KCP	N/A	N/A	0
B1	CICB	12/06/94	21.23.06	JJJ	N/A	N/A	0
B1	CICB	12/06/94	21.23.06	CATD	N/A	N/A	51
B1	CICB	12/06/94	21.23.07	CSNE	N/A	N/A	52
B1	CICB	12/06/94	21.23.08	CSGM	N/A	ZA00	53
B1	CICB	12/06/94	21.23.15	VGBR	N/A	N/A	55
B1	CICB	12/06/94	21.23.20	CSNE	N/A	N/A	56
B1 CICB 12/06/94 21.23.20							306

Changing Columns by Specifying Variables

To produce a flashback report with different columns than the default report, specify one or more variables with the FLASHBACK LIST command. A column will be created in the report for each variable you specify.

The following commands produce a report that displays all the components comprising transaction lifetime. Note that TRAN EXPC and TRAN CPU are not true components of transaction lifetime, because the times found in these variables are accounted for in other variables.

```
FLASHBACK LIST TIME TRANID TRAN NUM    AND,
                    TRAN LIFE    AND,
                    TRAN USER    AND,
                    TRAN CICS    AND,
                    TRAN I/OTIME AND,
                    TRAN WAIT    AND,
                    TRAN SUSP    AND,
                    TRAN WTR     AND,
                    TRAN RSCS    AND,
                    TRAN EXPC    AND,
                    TERM I/OTIME AND,
                    TRAN CPU

COUNT 20
END
RUN
```

The preceding commands generate a report like the following partial sample:

TIME	TRAN ID	TRAN NUMBER	TRAN LIFE	TRAN USER	TRAN CICS	TRAN I/O	TRAN WAIT	TRAN SUSP	TRAN WTR	TRAN RSCS	TRAN EXPC	TRAN TMIO	TRAN CPU
21.16.58	CSDA	19	0.016	0.000	0.000	0.000	0.000	0.000	0.001	0.000	0.001	0.000	0.011
21.21.52	CIRB	20	13.30	0.000	0.000	0.000	1.375	0.000	0.002	5.519	0.004	1.521	1.175
21.22.41	TCP	0	911.9	0.000	0.000	0.381	901.3	0.000	2.173	3.597	0.325	0.000	2.110
21.24.20	TCP	0	912.3	0.000	0.000	0.000	906.8	0.000	1.372	1.900	0.233	0.000	1.051
21.22.42	KCP	0	912.3	0.000	0.000	0.000	0.000	0.000	895.7	0.000	0.000	0.000	1.878
21.23.06	JJJ	0	937.1	0.000	0.000	0.000	0.418	934.7	0.001	0.000	0.001	0.000	1.942
21.23.06	CATD	51	6.529	0.000	0.000	0.000	0.124	0.000	0.009	2.579	0.003	0.000	0.326
21.23.07	CSNE	52	0.081	0.000	0.000	0.000	0.000	0.000	0.009	0.000	0.000	0.000	0.068
21.23.08	CSGM	53	0.264	0.000	0.000	0.000	0.000	0.000	0.002	0.139	0.001	0.017	0.010
21.23.15	VGBR	55	0.332	0.000	0.000	0.000	0.000	0.000	0.080	0.221	0.002	0.000	0.023
21.23.15	CSNE	56	0.013	0.000	0.000	0.000	0.000	0.000	0.002	0.000	0.000	0.000	0.002
21.23.15		306	335.8	0.000	0.000	0.035	164.5	84.97	81.76	1.269	0.052	0.140	0.782

Using the Date and Time Operands

To produce a FLASHBACK LIST report that has the date, time, or day in the first column, or the date and time in the first two columns, specify the DATE, TIME, DAY, or DATETIME operand with the FLASHBACK LIST command.

Because the width of the output page limits the number of columns that can appear in a report, you can use the DATE, TIME, DAY, or DATETIME operand only if you specify the variables to be included in your report and limit their number. The number of variables you can specify depends upon the width of the column required for each.

The following FLASHBACK LIST command is specified with the DATETIME operand. The COUNT command, specified with the SELECTED operand, limits the report to 15 transactions meeting either the FROM or the TO criterion specified.

```
FLASHBACK LIST DATETIME TRANID TRAN LIFE AND,
                        TRAN I/OTIME AND,
                        TRAN WAIT AND,
                        TRAN SUSP AND,
                        TERM I/OTIME AND,
                        TRAN CPU
FROM 12/06/98 21:00:00
TO 12/06/98 22:00:00
COUNT 15 SELECTED
END
RUN
```

The preceding commands generate a report like the following partial sample:

DATE	TIME	TRAN ID	TRAN LIFE	TRAN I/O	TRAN WAIT	TRAN SUSP	TRAN TMIO	TRAN CPU
12/06/98	21.16.58	CSDA	0.016	0.000	0.000	0.000	0.000	0.011
12/06/98	21.21.52	CIRB	13.30	0.000	1.375	0.000	1.521	1.175
12/06/98	21.22.41	TCP	911.9	0.381	901.3	0.000	0.000	2.110
12/06/98	21.24.20	TCP	912.3	0.000	906.8	0.000	0.000	1.051
12/06/98	21.22.42	KCP	912.3	0.000	0.000	0.000	0.000	1.878
12/06/98	21.23.06	JJJ	937.1	0.000	0.418	934.7	0.000	1.942
12/06/98	21.23.06	CATD	6.529	0.000	0.124	0.000	0.000	0.326
12/06/98	21.23.07	CSNE	0.081	0.000	0.000	0.000	0.000	0.068
12/06/98	21.23.08	CSGM	0.264	0.000	0.000	0.000	0.017	0.010
12/06/98	21.23.15	VGBR	0.332	0.000	0.000	0.000	0.000	0.023
12/06/98	21.23.41	VGBQ	19.33	0.000	0.000	0.000	18.40	0.151
12/06/98	21.28.18	VGBS	306.6	0.000	0.000	0.000	304.6	0.633
12/06/98	21.28.20	CSAC	0.647	0.000	0.000	0.000	0.000	0.015
12/06/98	21.28.23	END	0.486	0.000	0.000	0.000	0.000	0.232
12/06/98	21.28.23	CSNE	0.020	0.000	0.000	0.000	0.000	0.003
12/06/98	21.28.23		251.3	0.024	113.1	58.42	20.29	0.602

Using the RANGE Command to Limit Reports

Specify the RANGE command with the FLASHBACK LIST command to produce a report about only those resources that fulfill the specified range criteria. You can specify more than one RANGE command in a single report.

The following commands produce a report showing all transactions that had an average terminal response time greater than ten seconds:

```
FLASHBACK LIST
  RANGE TERM RESP 10+ DEFERRED
END
RUN
```

The preceding commands generate a report like the following partial sample:

CICS ID	END TIME	TERM ID	OPER ID	TRAN ID	TRAN NUMBER	TRAN LIFE	TRAN CPU	TERM RESP	TERM IO	TRAN I/O	FILE COUNT	FILE USE	PROG COUNT	ABEND CODE
B1	23.25.13	N/A	CSGM		276	56.41	0.007	28.19	2	0.000	0	0	1	
A1	23.25.02	N/A	CSGM		27	44.26	0.065	21.65	2	0.000	0	0	1	
A1	01.47.08	WAK	WORK		278	10.92	0.317	10.92	2	0.000	0	0	7	
A1	01.48.24	WAK	WORK		279	15.99	0.060	15.99	2	0.000	0	0	4	
B1	02.11.44	N/A	X25T		3498	14.20	0.178	14.20	2	0.106	1	1	8	
B1	02.14.43	AFL	RAPS		3708	13.18	0.062	13.18	2	0.000	0	0	4	
A1	02.23.35	WAS	WORK		794	24.26	0.059	24.26	2	0.000	0	0	4	
A1	02.41.37	AFK	DEAL		1157	22.84	0.654	16.20	3	0.339	2	114	5	
B1	02.41.48	N/A	X25T		5247	12.28	1.820	12.28	2	5.885	2	432	8	
A1	02.42.11	PSH	WARU		1164	36.29	5.162	14.60	4	0.000	0	0	7	
B1	02.44.43	N/A	X25T		5411	14.43	0.139	14.43	2	0.046	1	1	8	
B1	02.49.25	N/A	FXFR		5671	38.82	5.740	38.82	2	23.64	3	618	1	

FLASHBACK DETAIL Reports

The following section discusses operands and sample commands and provides sample output for FLASHBACK DETAIL reports.

FLASHBACK DETAIL Command

Use the FLASHBACK DETAIL command to produce a tabular report containing one page of output per transaction. The format of this page is fixed.

Syntax

The syntax of the FLASHBACK DETAIL command is as follows:

```
FLASHBACK DETAIL [ idoption(identifiers) ... ]
```

Operand

The following list describes the FLASHBACK DETAIL command operand:

Operand	Meaning
<i>idoption(identifiers)</i>	Any ID option followed by one or more identifiers. You can specify multiple ID options.

Report Format

Unless overridden by the COUNT command, end-of-file is forced after 500 transaction records are read. If the FROM and TO commands are specified, only records falling within the time period specified are counted.

Default FLASHBACK DETAIL Report

The following commands generate the default FLASHBACK DETAIL report:

```
FLASHBACK DETAIL
  FROM 12/09/98 00:08:00
  TO   12/09/98 00:08:10
END
RUN
```


Detail Headings

Detail Heading (Line 1)	Description
ID	Partition ID in which the data was collected
DATE	Date the transaction completed
TIME	Time the transaction completed
TERMID	Terminal ID
OPERID	Operator ID
TRANID	Transaction ID
NUMBER	Transaction number
C	Indicator of whether transaction is conversational: Y for YES or blank for NO
AB	Abend code, if any

Clock Information

Clock Information	Description
USER	User code time and percentage (version 4 only)
PGM	Program run time and percentage, not including waits
I/O	I/O wait time and percentage
WAIT	ECB wait time and percentage
SUSP	Suspend time and percentage
TMIO	Terminal I/O time and percentage
WTR	Waiting to run time and percentage
RSCS	Resource wait time and percentage
LIFE	Transaction lifetime and percentage
CPU	Transaction CPU time and percentage
DLI	DLI time and percentage

Terminal Information

Terminal Information	Description
RESPONSETIME	Average response time
MAX RESPONSETIME	Maximum response time
TOTAL READS	Total read count
AVE BYTES READ	Average read size
TOTAL WRITES	Total write count
AVE BYTES WRITTEN	Average write size

Storage Information

Storage Information	Description
TOTAL GETMAINS	Number of GETMAINS issued
TOTAL FREEMAINS	Number of FREEMAINS issued
STORAGE VIOLATIONS	Number of storage violations
TOTAL STORAGE	Total amount of GETMAINED storage
AVE GETMAIN SIZE	Average GETMAIN size
MAX GETMAIN SIZE	Maximum GETMAIN size
CURRENT STORAGE	Highest current storage usage
SUSPEND TIME	Suspend time spent waiting for a storage request

Dispatch Information

Dispatch Information	Description
DISPATCHES	Number of dispatches
SINGLE WAITS	Number of single waits
MULT WAITS	Number of multiple waits
CICS WAITS	Number of CICS waits
ENQUEUEES	Number of enqueuees
DEQUEUEES	Number of dequeuees
SUSPENDS	Number of suspends
SCP WAITS	Number of storage control program waits
TRP WAITS	Number of trace program waits
JCP WAITS	Number of journal control program waits
ICP WAITS	Number of interval control program waits
TSP WAITS	Number of temporary storage program waits
TDP WAITS	Number of transient data program waits
FCP WAITS	Number of file control program waits
DCP WAITS	Number of dump control program waits
PCP WAITS	Number of program control program waits
DLI WAITS	Number of DL/I program waits
BMS WAITS	Number of basic mapping support waits
TCP WAITS	Number of terminal control program waits
EIP WAITS	Number of temporary storage program waits
MRO WAITS	Number of multiple region option waits
SYS WAITS	Number of system waits
SOS WAITS	Number of short-on-storage waits
WRITE WAITS	Number of write waits

Temporary Storage Information

Temporary Storage Information	Description
TOTAL GETS	Number of temporary storage gets
TOTAL PUTS	Number of temporary storage puts
TOTAL PUTQS	Number of temporary storage putqs
PUTS TO MAIN STOR	Number of temporary storage puts to main storage
PUTS TO AUX STOR	Number of temporary storage puts to aux storage
AVE BYTES/GET	Average temporary storage get size
AVE BYTES/PUT	Average temporary storage put size
AVE TEMP I/O TIME	Average temporary storage I/O time
SUSPEND TIME	Suspend time spent waiting for a temp storage request

File Information

Each file used by the transaction is listed by name, and, if Unicenter CA-Explore is configured to collect file information by program, each program used is listed by name as well. The information shown in the following table is provided for each file and program listed. The total for each statistic for all files used is given under the heading TOTAL, with the exception of the statistics AVE REQ. TIME, I/O SERV. TIME and MAX I/O TIME, for which a total is inappropriate.

If more files meet the selection criteria specified than can be shown in the report, an asterisk (*) is placed just to the left of the column heading TOTAL. The resulting *TOTAL column lists the total of each statistic for all files, including the files not listed.

File Information	Description
TOT REQ. TIME	Total file request time
AVE REQ. TIME	Average file request time
TOT I/O TIME	Total I/O wait time
I/O SERV. TIME	Average I/O service time per request
MAX I/O TIME	Maximum I/O wait time
BUF WAIT TIME	Total VSAM buffer wait time
SOS WAIT TIME	Total short on strings wait time
PSOS WAIT TIME	Total pseudo short on strings wait time
WRITES	Total write count
READS	Total read count
READS FOR UPDT	Total reads for update count
ADDS	Total adds count
DELETES	Total deletes count
BROWSES	Total browses count
SOS WAITS	Number of short on strings waits
PSOS WAITS	Number of pseudo short on strings waits
SHR STR WAITS	Number of shared string waits
BUFFER WAITS	Number of VSAM buffer waits
XCL WAITS	Number of exclusive control waits
CI/CA SPLT WAITS	Number of CI/CA split waits

Program Information

Each program used by the transaction is listed by name, and the following information is provided for each. The total for each statistic appears in the row labeled *TOTAL*.

Program Information	Description
TIME	Time that the transaction spent executing program code. A program can link to other programs, and unless it performs an XCTL each time it links to another program, the time these additional programs are active is included in this statistic.
USE	Program use count
XCTLS	Program transfer control count
LOADS	Program load count
LINKS	Program link count
FETCHS	Program fetch count
ABENDS	Program abend count

User Exits

This appendix describes two user exits that are supplied with Unicenter CA-Explore for CICS.

Note: To defer the user exit call until after all selection criteria have been applied, and have the report writer pass to the user exit only those records passing all selection criteria (such as ID options and RANGE commands), use the DEFEREXT operand of the OPTION command, as explained in the chapter titled “Commands.”

User Exit Execution

The Unicenter CA-Explore for CICS report writer lets you specify a user exit routine. The report writer loads the exit routine during initialization, and then passes control to the routine according to the exit command you have specified.

Two user exit routines, EXPGUX and PRTGUX, are supplied with Unicenter CA-Explore for CICS. The following table lists the user exits, when the report writer passes control to each, and the exit command used to specify that routine:

User Exit	When Control Is Passed	Exit Command
EXPGUX	After each record is read and before the report writer processes the record	EXITRTN
PRTGUX	Before each line is printed	PRTEXIT

Source Code

The source code for the user exits is given in the following members in the Unicenter CA-Explore for CICS product library.sublibrary:

User Exit	Member Name
PRTGUX	PRTGUX.A
EXPGUX	EXPGUX.A

WARNING! *If you modify either of the supplied user exits, make a backup copy of the member using another name. When you install a new version of Unicenter CA-Explore for CICS, PRTGUX.A and EXPGUX.A will be overwritten. Unless you have a backup copy of the member, you will lose any modifications you made.*

Index

\$

\$ (dollar sign) marking syntax error, 6-2

*

* (asterisk)
comment line delimiter, 6-2
generic character in identifiers, 9-19
in output report, 8-3

+

+ (plus sign) generic character in identifiers, 9-19

-

- (not sign) generic character in identifiers, 9-19

A

ALONE operand, 6-66
PLOT command, 11-6
PLOT2 command, 11-10

AND operand, 11-23

Archive file
EXPCARC, 2-2
EXPCFBK, 2-2
filename for input, 6-45
JCL for reports, 3-3

Asterisk (*)
comment line delimiter, 6-2
generic character in identifiers, 9-19
in output report, 8-3

AVERAGE line, printing in reports, 6-53

AVERAGE operand, 6-53

AVERAGED operand, 6-21

B

Backing up user exits, A-2

BANNER parameter, 3-6

Blank lines
in job stream, 6-2
including in reports, 6-55

Blank spaces
in commands, 6-2
replacing blank identifiers in reports, 6-55
replacing zero values in reports, 6-58

BOTTOM operand, 10-22, 10-39

Braces, convention for use, 6-3

Brackets, convention for use, 6-3

C

Canned reports, 2-3, 5-1
CANRPT, 5-1
CICS.DCT, 5-12
CICS.DLIDBD, 5-13
CICS.DLIPGM6, 5-14
CICS.DLIPGM7, 5-15
CICS.DLIPSB6, 5-16
CICS.DLIPSB7, 5-18
CICS.FILESTAT, 5-20
CICS.FILEWAIT, 5-22

CICS.INTERVAL.STAT, 5-23
CICS.INTERVAL.STORAGE.STAT, 5-24
CICS.MAXTIME, 5-25
CICS.OPERSTAT, 5-27
CICS.PERFORMANCE.RECORD.
ANALYSIS.STAT, 5-29
CICS.PROGUSE, 5-33
CICS.REVIEW.STAT, 5-34
CICS.STORAGE, 5-36
CICS.SYSTEM.THRESHOLD.STAT, 5-38
CICS.TEMPSTOR, 5-39
CICS.TERMIO, 5-41
CICS.TERMRESP, 5-42
CICS.TERMRESP.RANGE, 5-44
CICS.TERMSTAT, 5-46
CICS.TERMTRAN, 5-48
CICS.TIMESTAT, 5-50
CICS.TRANFILE, 5-52
CICS.TRANFIO, 5-54
CICS.TRANGETM, 5-56
CICS.TRANLIFE.RANGE, 5-58
CICS.TRANPROG, 5-60
CICS.TRANRESP, 5-62
CICS.TRANSTAT, 5-64
CICS.TRANSTOR, 5-66
CICS.TRANTERM, 5-67
CICS.TRANTIO, 5-69
CICS.TRANUSE, 5-71
CICS.TRANWAIT, 5-72
CICSSTAT, 5-11
creating your own, 5-5, 5-6
descriptions of, 5-7
expansion, 5-2
generating, 5-1
how canned reports work, 6-7
tailoring, 5-3
VSE.CICS.STAT, 5-74

CANRPT command, 5-1
secondary commands, 5-3

CARDLEN command, 6-8

CASE operand, 6-53

CASE parameter, 3-6

CHAR1 operand, 11-8, 11-23

CHAR2 operand, 11-8

Characters, nonprintable, 6-54

Charge-back reports
COST command, 6-9
creating distributed cost report, 6-9
examples, 10-34

CICS and VSE systems, canned report, 5-74

CICS systems, canned report on activity, 5-11

CICS.DCT canned report, 5-12

CICS.DLIDBD canned report, 5-13

CICS.DLIPGM6 canned report, 5-14

CICS.DLIPGM7 canned report, 5-15

CICS.DLIPSB6 canned report, 5-16

CICS.DLIPSB7 canned report, 5-18

CICS.FILESTAT canned report, 5-20

CICS.FILEWAIT canned report, 5-22

CICS.INTERVAL.STAT canned report, 5-23

CICS.INTERVAL.STORAGE.STAT canned report, 5-24

CICS.MAXTIME canned report, 5-25

CICS.OPERSTAT canned report, 5-27

CICS.PERFORMANCE.RECORD. ANALYSIS.STAT
canned report, 5-29

CICS.PROGUSE canned report, 5-33

CICS.REVIEW.STAT canned report, 5-34

CICS.STORAGE canned report, 5-36

CICS.SYSTEM.THRESHOLD.STAT canned report, 5-38

CICS.TEMPSTOR canned report, 5-39

CICS.TERMIO canned report, 5-41

CICS.TERMRESP canned report, 5-42

CICS.TERMRESP.RANGE canned report, 5-44

CICS.TERMSTAT canned report, 5-46

CICS.TERMTRAN canned report, 5-48

CICS.TIMESTAT canned report, 5-50

CICS.TRANFILE canned report, 5-52

CICS.TRANFIO canned report, 5-54

CICS.TRANGETM canned report, 5-56

CICS.TRANLIFE.RANGE canned report, 5-58

CICS.TRANPROG canned report, 5-60

CICS.TRANRESP canned report, 5-62

CICS.TRANSTAT canned report, 5-64

-
- CICS.TRANSTOR canned report, 5-66
 - CICS.TRANTERM canned report, 5-67
 - CICS.TRANTIO canned report, 5-69
 - CICS.TRANUSE canned report, 5-71
 - CICS.TRANWAIT canned report, 5-72
 - CICSSTAT canned report, 5-11
 - Clock information, FLASHBACK DETAIL report, 13-11
 - Column headings, changing in tabular reports, 10-13
 - Comma, for continuing to new line, 6-2
 - Command expansions, 5-2
 - Commands
 - blank lines in job stream, 6-2
 - blank spaces in job stream, 6-2
 - CANRPT, 5-1
 - CARDLEN, 6-8
 - case, 6-2
 - COMMENT, 6-8
 - comment lines in job stream, 6-2
 - continuing to new line, 6-2
 - conventions for showing syntax, 6-3
 - COST, 6-9
 - COUNT, 6-12
 - DEFINE, 6-13
 - using with ID options, 9-22
 - descriptions, 6-1
 - EACH, 6-19
 - END, 3-8, 6-22
 - EXCLUDE, 6-23
 - EXITRTN, 6-24, A-1
 - FLASHBACK, 13-1
 - FLASHBACK DETAIL, 6-24, 13-9
 - FLASHBACK LIST, 6-25, 13-2
 - FROM, 6-27
 - GLOBAL, 6-31
 - GOTO, 6-32
 - GRAF, 6-33, 12-2
 - GRAF2, 6-35, 12-13
 - GROUP, 6-38
 - HEADER, 6-39
 - HPLOT, 6-41, 11-29
 - IF, 6-42
 - INCLUDE, 6-43
 - INPUT, 6-44
 - INVOICE, 6-47
 - line length, setting, 6-8
 - LINECOUNT, 6-50
 - listing before printing, 6-54
 - maximum number of input records, setting, 6-12
 - MPLOT, 6-50, 11-22
 - OPTION, 6-52
 - OUTPUT, 6-59
 - PERIOD, 6-61
 - PLOT, 6-63, 11-2
 - PLOT2, 6-64, 11-10
 - position in line, 6-2
 - processing order, 6-2
 - PRTEXT, 6-65, A-1
 - RANGE, 6-66
 - rules for entering, 6-2
 - RUN, 3-8, 6-68
 - SELECT, 6-68
 - setting with OPTION command, 6-54
 - SHIFT, 6-70
 - syntax conventions for presenting, 6-3
 - TAB2, 6-74, 10-36
 - TABulate, 6-71, 10-2
 - TITLE1, 6-75
 - TITLE2, 6-75
 - TITLE3, 6-75
 - TO, 6-27
 - VPLOT, 6-78, 11-13
 - VPLOT2, 6-79, 11-19
 - COMMENT command, 6-8
 - Comment line
 - delimiter, asterisk (*) as, 6-2
 - in output reports, 6-8
 - Compression, 5-24, 7-7, 7-57
 - Continuation character in commands, 6-2
 - Conventions for command syntax, 6-3
 - COST command, 6-9
 - DEFERRED operand, 6-9
 - examples, 10-34
 - SELECT operand, 6-9
 - COUNT command, 6-12
 - SELECTED operand, 6-12
 - COUNT variable option, 8-2
-
- ## D
-
- Database activity
 - DL/I, canned report, 5-13
 - DATAONLY operand, 6-53
 - DATAONLY parameter, 3-6

Dataset activity
 transient, canned report, 5-12
 VSAM, canned report, 5-20

DATASET variable class
 for INTerval record type, 7-5
 for PERFormance record type, 7-17
 for REVIEW record type, 7-57

Date and time
 actual, 6-28
 default, 6-27
 relative, 6-28

Date format, reports, 3-6

DATE operand, 10-7

DATEFRMT operand, 6-53

DATEFRMT parameter, 3-6

Dates
 excluding dates or date ranges, 6-23
 including column in tabular report, 10-5, 10-7
 including dates or date ranges, 6-43
 including in report titles, 6-76

DATETIME operand, 10-5

DAY operand, 10-7

Decimal positions in output fields, 8-2

Default value, convention for presenting, 6-3

DEFEREXT operand, 6-54

DEFERRED operand, 6-9, 6-66, 10-31

DEFINE command, 6-13
 using with ID options, 9-22

DELAY operand, 6-54

Detail headings
 FLASHBACK DETAIL report, 13-11

Diagnostic
 dump of input, 6-54
 tracing facility, 6-58

DIFF variable option, 10-14

Disk log file
 JCL for reports, 3-5

Disk space, minimizing use, 6-56

Disk space, minimizing use of
 CICS.PERFORMANCE.RECORD.
 ANALYSIS.STAT report, 5-29

Dispatch information
 FLASHBACK DETAIL report, 13-13

Distributed cost report, creating, 6-9

Distribution plot report (MPLOT), 11-22

DL/I
 database activity, canned report, 5-13
 program activity r6.x, canned report, 5-14
 program activity r7.x, canned report, 5-15
 program specification block (PSB) activity r6.x,
 canned report, 5-16
 program specification block (PSB) activity r7.x,
 canned report, 5-18

DLI variable class, Performance record variables
 BUFPERF, 7-21
 CHKP, 7-21
 DELETE, 7-21
 GETNEXT | GN, 7-21
 GETUNIQUE | GU, 7-21
 GHN, 7-21
 GHNP, 7-21
 GHU, 7-21
 GNP, 7-21
 INSERTs, 7-21
 MAXI/OTime, 7-21
 MAXREQtime, 7-21
 READs, 7-21
 REPLACEs, 7-22
 REQTIME, 7-22
 SERVETIME, 7-22
 SIOs, 7-22
 SIOTime, 7-22
 TYPE1, 7-22
 TYPE2, 7-22
 USE | USAGE | REQUESTs | IO, 7-22
 WRITE, 7-22

Dollar sign (\$) marking syntax error, 6-2

DSA variable class
 for REVIEW record type, 7-57

DUMP operand, 6-54

Dump, printing diagnostic dump of input, 6-54

E

EACH command, 6-19
 AVERAGED operand, 6-21
 examples with TAB command, 10-5

ECHO operand, 6-54

ECHO parameter, 3-6

Ellipsis, convention for use, 6-3

END command, 3-8, 6-22

End-of-file
 on I/O error, 6-55
 when maximum TO time is exceeded, 6-55

ERPT267 message, 6-58

ERPT274 message, 6-57

ERPTTAB macro, 5-6

Errors
 excluding messages from print and display, 6-58
 I/O, forcing end-of-file on, 6-55
 in syntax, dollar sign (\$) marking, 6-2

EXCLUDE command, 6-23
 ID operand, 6-23
 ID operand with SELECT option, 10-17

Excluding identifiers with generic character, 9-19

EXEC statement
 BANNER parameter, 3-6
 CASE parameter, 3-6
 DATAONLY parameter, 3-6
 DATEFRMT parameter, 3-6
 ECHO parameter, 3-6
 parameters, 3-5

EXIT variable class, 7-25

EXITRTN command, 6-24, A-1

Expansion, canned report, 5-2

EXPCARC archive file, 2-2

EXPCFBK flashback file, 2-2

EXPGUX user exit, A-1
 EXITRTN command, 6-24

EXPRDT phase, 5-5

F

FILE (DATASET) variable class
 for INTerval record type, 7-5
 for PERFormance record type, 7-17

File information
 FLASHBACK DETAIL report, 13-15

Files
 canned reports
 file use by transaction, 5-52, 5-54
 file waits, 5-22
 FILE (DATASET) variable class
 for INTerval record type, 7-5
 for PERFormance record type, 7-17

FLASHBACK commands, 4-18, 13-1

FLASHBACK DETAIL command, 4-19, 6-13, 6-24, 13-9

FLASHBACK DETAIL report, 13-9
 clock information, 13-11
 default, 13-9
 detail headings, 13-11
 dispatch information, 13-13
 file information, 13-15
 program information, 13-16
 storage information, 13-12
 temporary storage information, 13-14
 terminal information, 13-12

Flashback file
 filenames for input, 6-45
 JCL for reports, 3-2
 short-term data, 2-2

FLASHBACK LIST command, 4-18, 6-25, 13-2

FLASHBACK LIST report, 13-2
 date and time, 13-7
 default, 13-3
 ID options, 13-4
 limiting range, 13-8
 tailoring headings, 13-6

Flashback reports, 2-8, 13-1
 limiting output, 13-1

FOR operand, 10-20, 10-21, 10-37

FORMAT option for identifiers, 6-13

FORMAT variable option
using with ID options, 9-20
using with variables, 8-2

FROM command, 6-27
examples with TAB command, 10-5

G

Generic characters in identifiers
descriptions, 9-19
examples in TAB report, 10-19

GETMAINS by transaction, canned report, 5-56

GLOBAL command, 6-31

Global commands list, 6-4

GOTO command, 6-32

GRAF command, 4-16, 6-33, 12-2

GRAF report, 12-2
changing plot characters, 12-20
changing scale, 12-8
defining period covered, 12-10
for alternate resource, 12-5
for average lifetime, 12-12
for multiple periods, 12-11
limiting with FOR operand, 12-7
limiting with ID options, 12-6
simple, 12-4
sorting, 12-9

GRAF2 command, 4-17, 6-35, 12-13

GRAF2 reports, 12-13
for alternate resource, 12-18
limiting with FOR operand, 12-16
multiple-variable performance comparison, 12-19
scale, tailoring, 12-17

Graph reports, 2-7, 12-1

GROUP command, 6-38

H

HDR operand, 11-23

HDR1 operand, 10-13, 10-36, 11-23

HDR2 operand, 10-13, 10-36, 11-23

HEADER command, 6-39

Hexadecimal identifiers, 9-19

HEXID operand, 6-54

HIGHLIGHT operand, 6-67

History reporting
command syntax, 1-5
commands, 3-6
commands in job stream, 3-12
introduction, 1-1
multiple commands, 3-10
primary report commands, 3-7
reading reports, 4-1
secondary report commands, 3-8
tailoring reports, 3-10

Horizontal reports, 11-2, 11-29

HYPLOT command, 4-14, 6-41, 11-29
operands, 11-29
SCALE(*n*) operand, 11-31

I

I/O errors, end-of-file on, 6-55

I/Os, canned reports
terminal I/Os, 5-41
terminal I/Os, by transaction, 5-69
transient dataset I/Os, 5-12
VSAM dataset I/Os, 5-54

ID operand, 6-23, 6-67
of INCLUDE command, 6-43

ID options, 2-10, 9-1
defining, 6-13, 6-15
defining new, 9-22
definition, 7-1
descriptions, 9-2
formatting, 9-20
list, 9-2
PLOT report, 11-5
redefining, 9-22
specifying identifiers, 9-18, 9-19
using with
both commands and variables, 9-17
commands, 9-7
commands or variables--the difference, 9-15
variables, 9-14

ID1 operand, 10-17

Identifiers, 2-10

- assigning new values for, 6-42
- description, 9-18
- formatting, 6-13, 9-20
- generic characters, 9-19
- hexadecimal specification, 9-19
- INVALID (for nonprintable identifier), 6-54
- purpose, 9-18
- specifying with ID options, 9-18
- string printed when no identifier exists, 6-55
- type valid for each ID option, 9-2

IF command, 6-42

INCLUDE command, 6-43

- ID operand with SELECT option, 10-17

Input

- files, list, 6-45
- files, VSE, 6-46
- limiting records by date and time, 6-27
- printing dump of used or corrupted records, 6-54
- producing statistics on, 6-56
- using both VSE and CICS data, 11-2
- using records with negative values, 8-4
- zero values, including records with, 8-8

INPUT command, 6-44

INTerval record type

- DATASET variable class, 7-5
- description of, 7-2
- FILE (DATASET) variable class, 7-5
- list of variable classes, 7-3

INVALID, used for nonprintable identifiers, 6-54

INVOICE command, 6-47

IOERR operand, 6-55

IRC variable class, 7-26

ISC variable class, 7-28

J

JCL

- for reports using archive file, 3-3
- for reports using disk log file, 3-5
- for reports using flashback file, 3-2
- for reports using tape log file, 3-4

JOBZONE operand, 6-55

L

LINECOUNT command, 6-50

Log file, filenames for input, 6-45

LOGICAL variable option, 8-4

Long transactions, defined, 7-54

Lowercase

- convention for use, 6-3
- using in commands, 6-2

M

MAX variable option, 8-5

MAXDATE variable option, 8-6

MAXTIME variable option, 8-6

MAXTO operand, 6-55

Medium transactions, defined, 7-54

Messages

- ERPT267, 6-58
- ERPT274, 6-57
- excluding from print and display, 6-58
- status messages, 6-57

MIN variable option, 8-5

MONETARY operand, 6-55

MPLOT command, 4-11, 6-50, 11-22

- AND operand, 11-23
- CHAR1 operand, 11-23
- example, DATASET USE variable, 11-27
- example, multiple variables, 11-25
- example, one variable, 11-24
- HDR operand, 11-23
- HDR1 operand, 11-23
- HDR2 operand, 11-23
- operands, 11-23
- SCALE operand, 11-23
- using with HPLOT command, 11-32

MPLOT report, 11-22

N

n/a, printing instead of blank identifier, 6-55
Negative values, treating as unsigned, 8-4
Nonprintable characters, 6-54
Not sign (~) generic character in identifiers, 9-19
NULLID operand, 6-55
NULLLINE operand, 6-55
NXTIME variable option, 8-9
NXTND variable option, 8-10

O

Operator activity, canned report, 5-27
OPTION command, 6-52
 AVERAGE operand, 6-53
 CASE operand, 6-53
 DAFEREXT operand, 6-54
 DATAONLY operand, 6-53
 DATEFRMT operand, 6-53
 DELAY operand, 6-54
 DUMP operand, 6-54
 ECHO operand, 6-54
 HEXID operand, 6-54
 IOERR operand, 6-55
 JOBZONE operand, 6-55
 MAXTO operand, 6-55
 MONETARY operand, 6-55
 NULLID operand, 6-55
 NULLLINE operand, 6-55
 PERIOD operand, 6-55
 PGMSTOR operand, 6-55
 PRINT operand, 6-56
 PRINTLEN operand, 6-56
 PRODUCT operand, 6-56
 RANGE operand, 6-56
 RECAP operand, 6-56
 RECORD operand, 6-56
 RECSTAT operand, 6-56
 REWIND operand, 6-57
 RPTSTAT operand, 6-57
 STORAGE operand, 6-57
 STORSTAT operand, 6-57
 SUMMARY operand, 6-57
 TAB2SKIP operand, 6-57
 TRACE operand, 6-58

WEEK operand, 6-58
XMSG operand, 6-58
ZEROFLD operand, 6-58

Or-sign, convention for use, 6-3
OUTPUT command, 6-59

P

Parentheses, convention for use, 6-3
PerCenT variable option, 8-6
PERformance record type
 description, 7-2
 DLI variable class, 7-21
 EXIT variable class, 7-25
 IRC variable class, 7-26
 ISC variable class, 7-28
 list of variable classes, 7-3
 PROGram variable class, 7-31
 SECURITY variable class, 7-35
 STORAGE variable class, 7-38
 TABLE variable class, 7-39
 TRANsaction variable class, 7-47
 WAIT variable class, 7-56
PERIOD command, 6-61
PERIOD operand, 6-55
PERIOD: line in plot reports, 4-6
PGMSTOR operand, 6-55
PLOT command, 4-5, 6-63, 11-2
 ALONE operand, 11-6
 CHAR1 operand, 11-8
 CHAR2 operand, 11-8
 operands, 11-3
 RATE variable option, 11-9
 SCALE operand, 11-7
Plot report, 2-6, 4-5, 11-1
 reading, 4-7
 VPLOT reports, 4-8
PLOT report, 11-4
 ID options, 11-5
PLOT2 command, 4-7, 6-64, 11-10
 ALONE operand, 11-10
 RATE variable option, 11-12
 SCALE operand, 11-10
Plus sign (+) generic character in identifiers, 9-19

Primary report commands
 definition, 6-5
 list, 6-5

PRINT operand, 6-56

PRINTLEN operand, 6-56

PRODUCT operand, 6-56

Program activity r6.x
 DL/I, canned report, 5-14

Program activity r7.x
 DL/I, canned report, 5-15

Program information, FLASHBACK DETAIL report, 13-16

Program specification block (PSB) activity r6.x
 DL/I, canned report, 5-16

Program specification block (PSB) activity r7.x
 DL/I, canned report, 5-18

Program use
 by transaction, canned report, 5-60
 graph of, canned report, 5-33

PROGram variable class, 7-31

PRTEXIT command, 6-65, A-1

PRTGUX user exit, A-1

R

RANGE command, 6-66
 ALONE operand, 6-66
 DEFERRED operand, 6-66
 DEFERRED operand, examples, 10-31
 examples with CANRPT command, 5-4
 examples with TAB command, 10-29, 10-31
 HIGHLIGHT operand, 6-67
 ID operand, 6-67
 UNDERSLINE operand, 6-67

RANGE operand, 6-56

RANGE variable option, 10-32

RANGE: line in plot reports, 4-6, 6-56

RATE variable option, 11-9, 11-12
 description, 8-6
 examples, 8-7
 examples with TAB command, 10-16

RECAP operand, 6-56

RECORD operand, 6-56

Record types
 descriptions, 7-2
 INTERval, 7-5
 REVIEW, 7-57
 SUMmary, 7-60
 SYStem, 7-64
 variable classes, 7-3

RECSTAT operand, 6-56

Report definition table (RDT), 5-5, 5-6, 5-7

Report writer, 1-1
 commands and terms, 1-3
 global commands, 3-6
 introduction, 3-1
 multiple reports, 3-10
 report commands, 3-6
 tailoring reports, 3-10

Reports
 BANNER parameter, 3-6
 canned, 2-3, 5-1
 CASE parameter, 3-6
 DATAONLY parameter, 3-6
 DATEFRMT parameter, 3-6
 defining field length, 8-2
 defining information, 2-9
 Distribution plot (MPLOT), 11-22
 ECHO parameter, 3-6
 EXEC statement parameters, 3-5
 EXPCARC, 2-2
 EXPCFBK, 2-2
 flashback, 2-8
 Flashback, 4-18, 13-1
 FLASHBACK DETAIL, 4-19, 13-9
 FLASHBACK DETAIL, default, 13-9
 FLASHBACK DETAIL, reading, 4-20
 FLASHBACK LIST, 4-18, 13-2
 FLASHBACK LIST, date and time, 13-7
 FLASHBACK LIST, default, 13-3
 FLASHBACK LIST, ID options, 13-4
 FLASHBACK LIST, limiting range, 13-8
 FLASHBACK LIST, reading, 4-18
 FLASHBACK LIST, tailoring headings, 13-6
 formatting, 2-4
 GRAF, 4-16, 12-2
 GRAF, changing plot characters, 12-20
 GRAF, changing scale, 12-8
 GRAF, defining period covered, 12-10
 GRAF, for alternate resource, 12-5
 GRAF, for multiple periods, 12-11
 GRAF, limiting with FOR operand, 12-7

- GRAF, limiting with ID options, 12-6, 12-12
- GRAF, reading, 4-16
- GRAF, simple, 12-4
- GRAF, sorting, 12-9
- GRAF2, 4-17, 12-13
- GRAF2, for alternate resource, 12-18
- GRAF2, limiting with FOR operand, 12-16
- GRAF2, multiple-variable performance comparison, 12-19
- GRAF2, reading, 4-17
- GRAF2, tailoring scale, 12-17
 - graph, 2-7
 - Graph, 12-1
 - horizontal, 11-2, 11-29
 - HYPLOT, 4-14
 - HYPLOT, reading, 4-15
 - ID options, 2-10
 - identifiers, 2-10
 - including lines with no activity, 6-55
 - input, selecting, 3-2
 - JCL using archive file, 3-3
 - JCL using disk log file, 3-5
 - JCL using flashback file, 3-2
 - JCL using tape log file, 3-4
 - listing commands used, 6-54
 - long-term data, 2-3
 - minimal commands, 3-9
 - MPLOT, 4-11, 11-13, 11-22
 - MPLOT, reading, 4-11
 - planning, 2-1
 - plot, 2-6
 - PLOT, 11-4
 - plot reports, reading, 4-6
 - PLOT2, reading, 4-7, 4-9
 - printing zeros in, 6-58
 - reading, 4-1
 - short-term data, 2-2
 - specifying input files, 6-44
 - specifying output files, 6-59
 - tabular, 2-5, 4-1
 - tabular reports, headings, 4-1, 4-3, 4-4
 - tailoring, 2-10
 - titles, 6-75
 - variable options, 2-9
 - variables, 2-9
 - vertical, 11-13
 - vertical ellipses, 1-6
 - VPLOT, 11-13
 - VPLOT2, 4-9, 11-13
 - VPLOT2, reading, 4-11
 - writing, 3-1

- Response time by terminal, canned report, 5-42
- REVIEW record type
 - DSA variable class, 7-57
 - TERMINAL variable class, 7-58
 - THRESHOLD variable class, 7-58
 - TRANSACTION variable class, 7-59
- REWIND operand, 6-57
- Rewinding tape after input, 6-57
- RIGHT variable option, 10-18
- RPTSTAT operand, 6-57
- RUN command, 3-8, 6-68

S

- SCALE operand
 - MPLOT command, 11-23
 - of GRAF command, 12-8
 - PLOT command, 11-7
 - PLOT2 command, 11-10
 - VPLOT command, 11-16
- SCALE(COMMON) operand, VPLOT2 command, 11-21
- SCALE(*n*) operand, 11-31
- Secondary report commands
 - definition, 6-5
 - list, 6-6
- SECURITY variable class, 7-35
- SELECT command, 6-68
- SELECT operand, 6-9
- SELECT variable option, 10-17
- SELECTED operand, 6-12
- SELZERO variable option, 8-8
- SHIFT command, 6-70
- Short transactions, defined, 7-54
- SORT operand, 10-36
- Sorting
 - tabular reports, 10-24 to 10-28
- STACK operand, VPLOT command, 11-18
- Statistics on input data records, producing, 6-56

-
- Storage
 - acquiring above or below 16M line, 6-57
 - printing storage statistics, 6-57
 - requirements, 1-2
 - tracing storage requests, 6-58
 - use by report writer, displaying, 6-55
 - use by transactions, canned reports, 5-36, 5-66
 - Storage information, FLASHBACK DETAIL report, 13-12
 - Storage limits, 7-9
 - STORAGE operand, 6-57
 - STORAGE variable class, 7-38
 - STORSTAT operand, 6-57
 - Subpool, 7-7
 - Summary line in tabular reports, 6-56, 6-57
 - SUMMARY operand, 6-57
 - SUMmary record type
 - description, 7-2
 - list of variable classes, 7-3
 - TERMinal variable class, 7-60
 - TRANsaction variable class, 7-61
 - Syntax
 - conventions for presenting commands, 6-3
 - errors, marked with dollar sign, 6-2
 - System activity, canned reports
 - VSE and CICS, 5-74
 - SYStem record type
 - description, 7-2
 - list of variable classes, 7-3
 - THRESHOLD variable class, 7-64
- ## T
-
- TAB command
 - BOTTOM operand, 10-22
 - DATE operand, 10-7
 - DATETIME operand, 10-5
 - DAY operand, 10-7
 - FOR operand, 10-9, 10-20, 10-21
 - HDR1 operand, 10-13
 - HDR2 operand, 10-13
 - TIME operand, 10-6, 10-9
 - TOP operand, 10-22
 - TAB2 command, 6-74
 - BOTTOM operand, 10-39
 - FOR operand, 10-37
 - HDR1 operand, 10-36
 - HDR2 operand, 10-36
 - reading TAB2 reports, 4-4
 - SORT operand, 10-36
 - syntax, 10-36
 - TOP operand, 10-39
 - TAB2SKIP operand, 6-57
 - TABLE variable class, 7-39
 - Tabular reports, 2-5, 10-1
 - complex reports, 4-3
 - simple reports, 4-2
 - sorting, 10-24 to 10-28
 - TABulate command, 6-71, 10-2
 - syntax, 10-2
 - Tape log file
 - JCL for reports, 3-4
 - Tape, rewinding after input, 6-57
 - Temporary storage information
 - FLASHBACK DETAIL report, 13-14
 - Temporary storage use, canned report, 5-39
 - Terminal information
 - FLASHBACK DETAIL report, 13-12
 - TERMinal variable class
 - for REVIEW record type, 7-58
 - for SUMmary record type, 7-60
 - Terminals
 - activity, canned reports
 - by maximum response time, 5-46
 - by transaction, 5-67
 - I/Os, 5-41
 - with subgraphs of transaction use, 5-48
 - response time, canned reports
 - by terminal, 5-42
 - by transaction, 5-62

Terminology

- global commands, 1-3
- ID options, 1-4
- identifiers, 1-4
- operands, 1-3
- primary report commands, 1-3
- report commands, 1-3
- report writer commands, 1-3
- secondary report commands, 1-3
- variable options, 1-4
- variables, 1-3

THRESHOLD variable class

- for REVIEW record type, 7-58
- for SYStem record type, 7-64

Thresholds, canned report, 5-38

Time

- defining shifts, 6-70
- defining time periods, 6-61
- excluding time ranges, 6-23
- format for output
 - hhh:mm:ss*, 8-9
 - hhh:mm:ss:lll*, 8-9
 - ssss.lll*, 8-9
- including column in tabular report, 10-5, 10-6
- including in report titles, 6-76
- including time ranges, 6-43
- zone, setting, 6-55

TIME operand, 10-6, 10-9

TITLE n commands, 6-75

TO command, 6-27

- examples with TAB command, 10-5
- maximum TO time checking, 6-55

TOP operand, 10-22, 10-39

TOTAL variable option, 8-8

Trace (diagnostic tracing facility), 6-58

TRACE operand, 6-58

TRANsaction variable class

- for PERFormance record type, 7-47
- for REVIEW record type, 7-59
- for SUMmary record type, 7-61

Transactions

- CICS, daily summary, 5-34
- defining long, medium, and short, 7-54

Transactions, canned reports

- activity
 - by lifetime, 5-58
 - by maximum response time, 5-64
 - by operator, 5-27
 - by terminal, 5-48
 - by terminal response time, 5-44
 - by transaction use and program use, 5-60
 - by transaction use and terminal use, 5-67
- file use, 5-52
- GETMAINs, 5-56
- lifetime, 5-50
- lifetime and terminal response time, 5-62
- maximum lifetime, 5-25
- storage activity, 5-36
- storage use, 5-66
- temporary storage use, 5-39
- usage and terminal I/O, 5-69
- usage graph, 5-71
- VSAM dataset use, 5-54
- wait time, 5-72

Transient datasets

- canned report on I/Os, 5-12

U

UNDERLINE operand, 6-67

Underscoring, convention for use, 6-3

Uppercase

- convention for use, 6-3
- forcing with CASE operand, 6-53
- using in commands, 6-2

User exits, A-1

- defining exit to be called
 - PRTEXIT command, 6-65
- EXITRTN command, 6-24
- EXPGUX, A-1
- processing order, 6-54, A-1
- PRTGUX, A-1
- source code, A-2
- when to back up, A-2

Using CICS and VSE variables, 6-46

V

Variable options, 8-1

- COUNT, 8-2
- definition, 7-1
- DIFF, 10-14
- FORMAT
 - using with ID options, 9-20
 - using with variables, 8-2
- LOGICAL, 8-4
- MAX, 8-5
- MAXDATE, 8-6
- MAXTIME, 8-6
- MIN, 8-5
- NXTIME, 8-9
- NXTND, 8-10
- PerCenT, 8-6
- RANGE, 10-32
- RATE, 8-6, 10-16
- RIGHT, 10-18
- SELECT, 10-17
- SELZERO, 8-8
- syntax, 8-2
- TOTAL, 8-8
- XTIME, 8-9
- XTIMEM, 8-9
- XTND, 8-9

Variables

- changing from count to rate, 10-16
- classes
 - DATASET
 - for INTerval record type, 7-5
 - for PERFormance record type, 7-17
 - for REVIEW record type, 7-57
 - DSA for REVIEW record type, 7-57
 - EXIT, 7-25
 - FILE (DATASET)
 - for INTerval record type, 7-5
 - for PERFormance record type, 7-17
 - IRC, 7-26
 - ISC, 7-28
 - list, 7-3
 - PROGram, 7-31
 - SECURITY, 7-35
 - STORAGE, 7-38
 - TABLE, 7-39
 - TERMinal
 - for REVIEW record type, 7-58
 - for SUMmary record type, 7-60
 - THRESHOLD
 - for REVIEW record type, 7-58
 - for SYStem record type, 7-64

TRANsaction

- for PERFormance record type, 7-47
- for REVIEW record type, 7-59
- for SUMmary record type, 7-61
- WAIT, 7-56
- convention for presenting, 6-3
- DATASET USE, 11-27
- defining, 6-13, 6-14, 7-1
- digits in output, 8-9, 8-10
- forming, 7-2
- negative
 - including in input, 8-4
 - treating as unsigned, 8-4
- parts, 7-2
- printing
 - as a maximum or minimum value, 8-5
 - as a percentage, 8-6
 - as a rate, 8-6
 - as a total, 8-8
 - date of highest value, 8-6
 - fewer digits, 8-10
 - more digits, 8-9
 - time of highest value, 8-6
 - zero values, 6-58
- time format of
 - hhh:mm:ss*, 8-9
 - hhh:mm:ss:lll*, 8-9
 - ssss.lll*, 8-9
- zero values, including records with, 8-8

Vertical bar (or-sign), convention for use, 6-3

Vertical reports, 11-13

- VPLOT command, 4-8, 6-78, 11-13
 - operands, 11-14
 - SCALE operand, 11-16
 - STACK operand, 11-18
 - using with HPLOT command, 11-30

- VPLOT2 command, 4-9, 6-79, 11-19
 - operands, 11-19
 - SCALE(COMMON) operand, 11-21

VSAM

- archive file EXPCARC, 2-2
- dataset activity, canned report, 5-20
- file activity, canned report, 5-20
- file waits, canned report, 5-22
- flashback file EXPCFBK, 2-2

VSE and CICS systems, canned report, 5-74

VSE.CICS.STAT canned report, 5-74

W

WAIT variable class, 7-56

Waits, canned reports
by transaction, 5-72
by VSAM files, 5-22

WEEK operand, 6-58

X

XMSG operand, 6-58

XTIME variable option, 8-9

XTIMEM variable option, 8-9

XTND variable option, 8-9

Z

Zeroes

in input, including records with, 8-8
including lines with no activity, 6-55
printing in reports, 6-58

ZEROFLD operand, 6-58

Zone (time zone), 6-55